THE IMPLEMENTATION OF A FINITE ELEMENT COMPUTER CODE AND ASSOCIATED PRE-AND POSTPROCESSOR INTO AE4101 AND AE4102 (FLIGHT VEHICLE STRUCTURAL ANALYSIS I AND II)

Dennis M. Losh

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

THE IMPLEMENTATION OF A FINITE ELEMENT COMPUTER CODE AND ASSOCIATED PRE-AND POSTPROCESSOR INTO AE4101 AND AE4102 (FLIGHT VEHICLE STRUCTURAL ANALYSIS I AND II)

by

Dennis M. Losh

December 1976

Thesis Advisor: Robert E. Ball

Approved for public release; distribution unlimited.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) The Implementation of a Finite Element Computer Code and Associated Pre-and Postprocessor Into AE4101 and AE4102 (Flight Vehicle Structural Analysis I and II)	5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; December 1976
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Dennis M. Losh	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	12. REPORT DATE December 1976
	13. NUMBER OF PAGES 91
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The objectives of the project described in this thesis were to: 1) provide the documentation that is needed for a Naval Postgraduate School student to use the general purpose finite element computer program called SAP IV, and 2) to make available, and prepare the users manual for a pre-and postprocessor program called SUBROUTINE PSAP. This subroutine, which was developed at the NASA Langley Research Center, has been modified to specifically plot the finite element model geometry for SAP IV models and to postprocess

20. (continued)

displacement data for those models on the NPS Calcomp Model 765 Plotter. The input and output for SAP IV and SUBROUTINE PSAP are discussed in detail. The codes have been used successfully in AE 4102, Flight Vehicle Structural Analysis II.

THE IMPLEMENTATION OF A FINITE ELEMENT COMPUTER CODE AND ASSOCIATED PRE-AND POSTPROCESSOR INTO AE4101 AND AE4102 (FLIGHT VEHICLE STRUCTURAL ANALYSIS I AND II)

by

Dennis M. Losh
Lieutenant
B.S.,United States Naval Academy,1970

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

December 1976

ABSTRACT

The objectives of the project described in this thesis were to : 1) provide the documentation that is needed for a Naval Postgraduate School student to use the general purpose finite element computer program called SAP IV, and 2) to make available, and prepare the users manual for, a pre-and postprocessor program called SUBROUTINE PSAP. This subroutine, which was developed at the NASA Langley Research Center, has been modified to specifically plot the finite element model gecmetry for SAP IV models and to postprocess displacement data for those models on the NPS Calcomp Model 765 Plotter. The input and output for SAP IV and SUBROUTINE PSAP are discussed in detail. The codes have been used successfully in AE 4102, Flight Vehicle Structural Analysis II.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

I wish to acknowledge the help that I have received in the course of my research.

I am particularly indebted to Professor R. E. Ball, who provided me with the guidance and aid necessary to complete the project.

I am also grateful to Professor Gilles Cantin, who provided me with valuable information in starting and carrying through my work.

I also wish to give credit to the NPS programming consultants, and in particular Sharon D. Raney, for the fine assistance I received in debugging my computer runs.

My gratitude is also heartfully extended to Richard Citerley for providing me with the computer card decks necessary to begin my project.

I especially want to express my sincere appreciation to my wife, Karen, for her patience and perseverance with the typing of this thesis.

I. INTRODUCTION

A. STRUCTURAL ANALYSIS SOFTWARE PACKAGES

The past decade has seen great strides in the field of computer software packages that have been developed for use in structural analysis. A great deal of time and effort has been expended in the development and distribution of these packages. Today there are structural analysis computer programs for almost every conceivable structure an engineer could desire to analyze. The types of analyses performed by a given package vary widely and may include features for linear or nonlinear materials, static analysis, dynamic analysis, buckling analysis, and nonlinear dynamic analysis, to name only a few. These programs can be grouped into two major categories - special purpose or general purpose.

In order to expose the students at the Naval Postgraduate School to the use of a general purpose structural analysis program, as well as to provide the capability of using such a program in research work, the finite element structural analysis program SAP IV (Reference 1) was acquired and made operational at the Naval Postgraduate School by Professor Gilles Cantin of the Mechanical Engineering Department. SAP IV can perform linear static and dynamic analyses on one-, two-, and three-dimensional structures.

B. PRE-AND POSTPROCESSORS FOR STRUCTURAL ANALYSIS PROGRAMS

After the development of a large number of structural analysis programs, users began to recognize that a disparity existed between efficient general purpose structural analysis programs and optimum utilization of these programs. Many of today's software structural analysis packages, such as SAP IV, require the user to prepare and reduce tremendous amounts of data. The need existed for some aids in processing and reducing these large quantities of data. Consequently, there have been many pre- and postprocessors developed for a specific use as well as for a general use basis during the past several years. The value of a given processing package lies in its ability to aid the user in preparing his model geometry, in his data checks, and in processing the output in an easily understood fashion. One of the most effective means utilized in preparing or reducing data is through the use of visual displays, whether they be designed primarily for graphic presentations, such as the Strcmberg-Carlson, or for paper plots, such as Calcomp. This ability to visually display input and output data is a highly valuable tool for the structural analyst.

C. THESIS MOTIVATION

The desire to have a pre- and postprocessor that could be used in conjunction with the general purpose structural analysis program SAP IV prompted the acquisition and implementaticn of a general use plotting package by this author. After researching the possible options, an existing program was obtained from Anamet Laboratories, San Carlos,

California. The program was originally developed at the Langley Research Center, Hampton, Virginia by Gary L. Giles for use with modern digital computers. The program, details of which can be found in Reference 2, generates oblique orthographic projections of three-dimensional finite element models and is distinguished by its provisions for generality, ease of use, different display options, and computational speed. The computer code was written for use on CDC 6000 series machines and had to be modified somewhat for use on the NPS 360/67. The modified version of the program is now available for use with the NPS Calcomp Plotter Model 765.

The primary purpose of this thesis is to provide the necessary documentation in order that students enrolled at the Naval Postgraduate School, and specifically in the courses AE 4101, 4102 (Flight Vehicle Structural Analysis I and II respectively), can, with a minimum of difficulty, effectively utilize SAP IV and its now-functional pre-and postprocessor PSAP. The remainder of this thesis is broken down into two major subdivisions,

1) II. <u>GUIDE TO THE USE OF SAP IV AT THE NAVAL POSTGRADUATE SCHOOL</u>

2) III. <u>GUIDE TO THE USE OF SUBROUNTINE PSAP, A PRE-AND POSTPROCESSOR FOR SAP IV</u>

Appendices A and B of this work provide detailed examples for the input preparation and output reduction of data using both SAP IV and SUBROUTINE PSAP.

II. GUIDE TO THE USE OF SAP IV AT NPS

A. DESCRIPTION OF SAP IV

SAP IV is a general purpose structural analysis digital computer program that can provide a finite element solution for both the static and dynamic analysis of linear structural systems. A detailed user's manual is contained in Reference 1. The program has the capacity to analyze very large three-dimensional systems, as well as small systems, with no loss in efficiency. SAP IV, which is coded in FORTRAN IV , is a very flexible program and can be considered a very efficient aid to the analyst. The purpose of this section is to provide the necessary additional documentation, above that provided in Reference 1, for a student at the Naval Postgraduate School to make use of the program.

The methods of analysis and the construction of the program are not included in this section, but can be found in Reference 1. The program contains nine finite elements of the following types:

(a) three-dimensional truss element,

(b) three-dimensional beam element,

(c) plane stress and plane strain element,

(d) two-dimensional solid element,

(e) three- dimensional solid element,

(f) variable-number-nodes thick shell and three-dimensional element,

(g) thin plate or thin shell element,

(h) boundary element,

(i) pipe element (tangent and bend).

There are numerous options and combinations of static and dynamic analysis that are available to the user of the program. Reference 1 provides specific details of the many available user options.

B. COMPUTER CARD DECK PREPARATION AT NPS FOR SAP IV

Figure 1 outlines the overall computer card deck necessary to access and utilize SAP IV as it is currently operational at NPS. A complete detailed breakdown of the necessary IBM job control cards follows on the next page.

```
// ( STANDARD GREEN JOB CARD )
//GO EXEC PGM=SAP,REGION=260K
//STEPLIB DD UNIT=2321,VOL=SER=CEL002,DISP=SHR,
//   DSN=F0559.SAPLM
//GO.FT01F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT02F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT03F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT04F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT05F001 DD DDNAME=SYSIN

//GO.FT06F001 DD SYSOUT=A,SPACE=(CYL,(3,1)),
//   DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)
//GO.FT07F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT08F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT09F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT10F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//   DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT11F001 DD SYSOUT=B,SPACE=(TRK,(20,2)),
//   DCB=(RECFM=FB,LRECL=80,BLKSIZE=7200)
//GO.SYSIN DD *


    ( SAP IV ---  DATA )


/ ( STANDARD NPS EOF CARD )
```

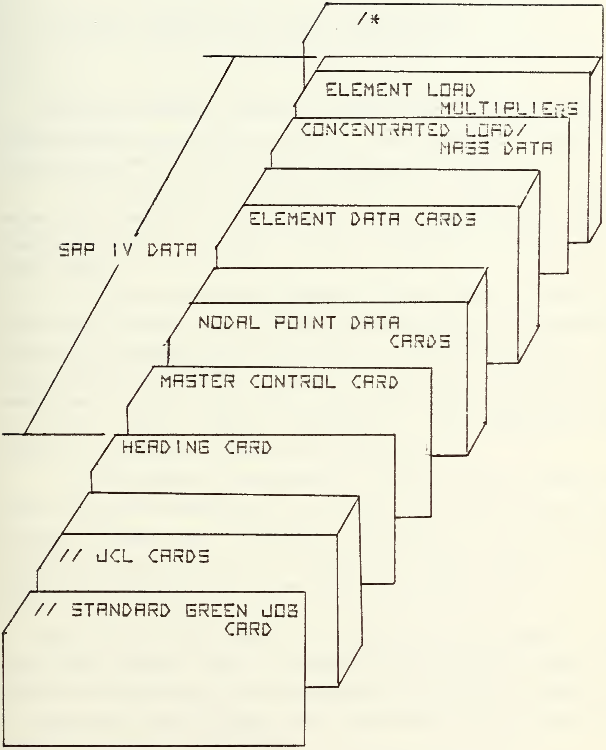


Figure 1 - SAP IV DECK SET-UP

After the //GO.SYSIN DD * card, the deck of SAP IV data that is prepared according to the instructions in the appendices of Reference 1 follows. Following the SAP IV data is the standard NPS end-of-file card (/*).

C. HELPFUL POINTS ON DATA DECK PREPARATION

There are several possible areas where a user can err in his preparation of data for SAP IV. The following items are representative of a few common sources of error.

1) Particular attention should be directed toward using the correct formats and correct card columns in key punching data on cards. (i.e., integer formats, right justified)

2) The program has several internal data generation features inherent in it. Simply because data were generated during the program execution does not necessarily mean they were generated correctly. Errors in the user- prepared input cards used in data generation can cause severe discrepancies to occur during program execution. The generated data should be carefully checked for accuracy.

3) In order to terminate a given problem a number of blank cards are necessary at the end of the input data deck. Specific points of interest pertaining to the problem of termination can be found in Reference 1 under note (1) on page II.1 and in note (1) on the top of page V.2.

SAP IV has a wide range of options where a clever user can fully exploit the full capabilities of the program. The detailed decription of these features is found in Reference 1; however, the following list summarizes a few of the more useful options and features.

1) Data check only mode of execution.

2) Nodal point and element data generation capabilities.

3) Five different types of analysis.

a. static analysis

b. eigenvalue /vector solution

c. forced dynamic response by mode superposition

d. response spectrum analysis

e. direct step by step integration

4) Automatic punched computer card output of displacement results. This feature was added to the program by this authcr to make possible graphic postprocessing.

The foregoing discussion is not intended to be a complete diagnostic summary of SAP IV, but rather an aid to the student who desires to get started using the program as it currently exists on the NPS IBM 360/67. The above discussion, along with a copy of Reference 1, should provide a jumping off place for a novice structural analyst. A complete example of the job control cards and input data deck for a static truss analysis can be found in Appendix A.

D. ALTERATION OF SAP IV AT NPS

Because SAP IV is a very flexible program, it is possible for a user to make modifications to the basic program. The computer code is complex, but it is not overly difficult to modify parts of the program in order to satisfy a specific user need. A method that has proven successful for this author, in the modification of a SAP IV subroutine to provide punched output of displacement data, and to create a personal version of SAP IV, is outlined in this section.

The first step is to define the subroutine name that the user wishes to modify. A listing of the source program is available through Professor R. E. Ball, Department of Aeronautics, or Professor G. Cantin, Department of Mechanical Engineering. Having defined those portions of the program for which modification is desired, the next step is to obtain a punched copy of those desired routines by using the following format.

```
// (STANDARD NPS JOB CARD)
//SYSPRINT DD SYSOUT=A,SPACE=(TRK,(10,1))
//SYSUT1 DD DISP=SHR,UNIT=2321,VOL=SER=CEL001,
// DSN=F0099.SAPSR
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
   PUNCH TYPORG=PO,MAXNAME=2
   MEMBER NAME=(SUBROUTINE NAME)
/*
```

NOTE: SUBROUTINE NAME is the name of the SAP IV subroutine required.

With the desired routine decks now in hand, the necessary changes can be incorporated into the subroutine deck and an object deck is then obtained as follows:

```
// (STANDARD NPS JOB CARD)
// EXEC FORTCD
// FORT.SYSIN DD *
(MODIFIED FORTRAN SOURCE DECK)
/*
```

The next and final step is to take the object deck obtained in the previous step and insert it into the proper position in the control cards that are illustrated on the following page.

```
// (STANDARD GREEN JOB CARD)
// EXEC LGO,PARM.LINK='OVLY,XREF,LIST'
//LINK.SYSLMOD DD DSN=F0559.SAPLM(SAP),UNIT=2321,VOL=SER=CEL002,
//    DISP=(NEW,KEEP),SPACE=(CYL,(35,2,2)),LABEL=RETPD=90
//LINK.SYSUT1  DD  SPACE=(CYL,(2,1))
//LINK.LIBRARY DD DSN=F0099.SAPLM,UNIT=2321,VOL=SER=CEL001,DISP=SHR
//LINK.SYSIN DD *

    ( MODIFIED FORTRAN OBJECT DECK )

    INCLUDE LIBRARY(SAP)
       ENTRY MAIN
   OVERLAY A
   INSERT ADDSTF,BOUND,CLAMP,INL,RUSS,SESOL,TRUSS
   OVERLAY A
   INSERT BEAM,NEWBM,SLAVE,TEAM,NEWB
   OVERLAY A
   INSERT CROSS,DOT,ELAW,FORMB,PLANE,PLNAX,POSINV,QUAD,VECTOR
   OVERLAY A
   INSERT BRICK8,DERIV,LOAD,LOSTR,PRIST,THREED
   OVERLAY A
   INSERT CSTSTR,LCTMOM,LCT9ST,LSTSTR,QDCOS,QTSHEL,SHELL,SLCCT,SLST,     X
                STRETR,TDCOS,TPLATE,TRFPRD,QTSARG,TRIARG,TRANSF
   OVERLAY A
   INSERT CROSS2,DER3DS,FACEPR,FNCT,INP21,SOL21,SSLAW,ST8R21,THDFE,      X
                VECTR2,GAUSS
   OVERLAY A
   INSERT BENDDC,BENDKS,PINVER,PIPE,PIPEK,PIPES2,PIPES3,SELECT,          X
                TANGDC,TANGKS,PIPEC
   OVERLAY A
   INSERT BANDET,DECOMP,EIGSOL,INVECT,JACOBI,MODES,REDBAK,SBLOCK,        X
                SCHECK,SECNTD,SOLEIG,SSPCEB
   OVERLAY A
   INSERT DISPLR,DISPLY,ELOUTH,EMID,GMTN,HISTRY,LOAD1,LOAD2,PPLOT,       X
                RESPON,STRSD1
   OVERLAY A
   INSERT ELOUTR,EMIDR,RESPEC,SD,SPECTR,STRESR
   OVERLAY A
   INSERT ADDMAS,ELOUTS,EMIDS,GROUND,INCLY,INOUT,INTHIS,LOADV,           X
                PLOAD,REDVK,SDSPLY,SOLSTP,SPLOT,STEP,TRIFAC
   NAME SAP
/ ( STANDARD NPS EOF CARD )
```

There are many options available in the creation of load module libraries and they are discussed in detail in Reference 4, sections II. and III. The portion of the preceding example control card deck that would necessitate modification is the //LINK.SYSLMOD card, where F0559.SAPLM(SAP) should be changed according the following format :

General format: Lnnnn. anyname

where L is :

S-student data set

F-faculty data set

nnnn is :

user number assigned

anyname is :

any unique name assigned by the user (1-6 characters in length with the first character alphabetic), and CEL002, should reflect an appropriate data cell with available space.

The procedure outlined above will create a modified version of SAP IV on a chosen data cell available to the user for a period of 90 days. The program can now be executed with appropriate modifications to the //STEPLIB card as discussed previously in section II. B. Any questions concerning the creation of load module libraries or their execution can be answered by any of the programing consultants on the first floor of Ingersoll Hall.

III. GUIDE TO THE USE OF SUBROUTINE PSAP, A PRE-AND POSTPROCESSOR FOR SAP IV

A. SUBROUTINE DESCRIPTION

SUBROUTINE PSAP is a modified version of the oblique orthographic projection program that is found in Appendix B of Reference 2. The program, originally developed for use at the NASA Langley Research Center, Hampton, Virginia, required some changes so that it could be used in conjunction with SAP IV and the NPS Model 765 Calcomp Plotter. The original version of the plotting package allowed for various geometry and displacement data input options. However, the subroutine, as currently filed, is constructed strictly for use with the input and output of SAP IV. With some slight modifications to the subroutine, it could be adapted to any number of different types of input geometry or displacement data decks. A method for accomplishing this is outlined in section III. C.

The current capability for generating oblique orthographic projections of SAP IV finite element models is limited to the following types of elements:

1) Type 1, Three-Dimensional Truss Elements
2) Type 2, Three-Dimensional Beam Elements
3) Type 3, Plane Stress Membrane Elements
4) Type 4, Two-Dimensional Finite Elements
5) Type 6, Plate and Shell Elements (Quadrilateral).

The undeformed topology of the finite element model, useful in checking input data, as well as the displaced topology

projection of the same model, can be obtained from PSAP. The subroutine contains many different options and permutations of those options, some of which are listed below.

1) plots of models annotated with grid point numbers of element numbers
2) plots of portions of models
3) exploded plots of model sections (i.e., line elements coincident with the edges of triangular or quadrilateral elements may be difficult to single out; program provides a capability to separate elements so that their absence or presence is easily detectable)
4) displacements superimposed on grid point coordinates of the undeformed structure
5) displacements represented as vectors extending the undisplaced grid points.

There are many more combinations of options that are available and will be more specifically outlined in the remainder of this section. Appendix B details a complete input-output example of a pin-jointed truss.

B. SUBROUTINE PSAP USE

1. General Set-up of Input Deck

In general, the correct sequence of computer cards required to utilize SUBROUTINE PSAP is shown schematically in Figure 2 and consists of eight separate major groups as follows:

1) a group of JCL cards and main program to allocate storage
2) a single card containing title information
3) Namelist OPTION containing values to determine if

proper storage allocation is available and specifying various program options

4) a geometry deck (SAP IV data deck) containing grid points and connectivity of the model

5) an optional single title card used to identify the deck of displacement data to be plotted

6) a single card containing the value of the total number of SAP IV load cases and an optional scale factor

7) the deck of displacement data to be plotted (output of the execution of SAP IV-static analysis)

8) Namelist PICT containing values to specify the type of plot desired and what information is to be included on the plots.

By repeating parts of the basic input data to the program, different plots of the same data can be generated.

2. Job Control Cards

The sequence of cards on the following page depicts the necessary JCL to execute SUBROUTINE PSAP.

```
// (STANDARD GREEN JOB CARD)
// EXEC FORTCLGP,REGION.GO=180K
//FORT.SYSIN DD *

   ( MAIN PROGRAM )

/ ( STANDARD NPS EOF CARD )
//LINK.USDD DD UNIT=2321,VOL=SER=CEL002,DISP=SHR,
// DSN=F0559.PSAPLM
//LINK.SYSIN DD *
     INCLUDE USDD(PSAP)
     ENTRY MAIN
/ ( STANDARD NPS EOF CARD )
//GO.FT10F001 DD UNIT=SYSDA,
// SPACE=(CYL,(3,1)),
// DCB=(RECFM=VS,BLKSIZE=3520)
//GO.SYSIN DD *

   ( DATA --- STARTING WITH THE TITLE CARD --- SEE FIGURE 2)

/ ( STANDARD NPS EOF CARD )
```

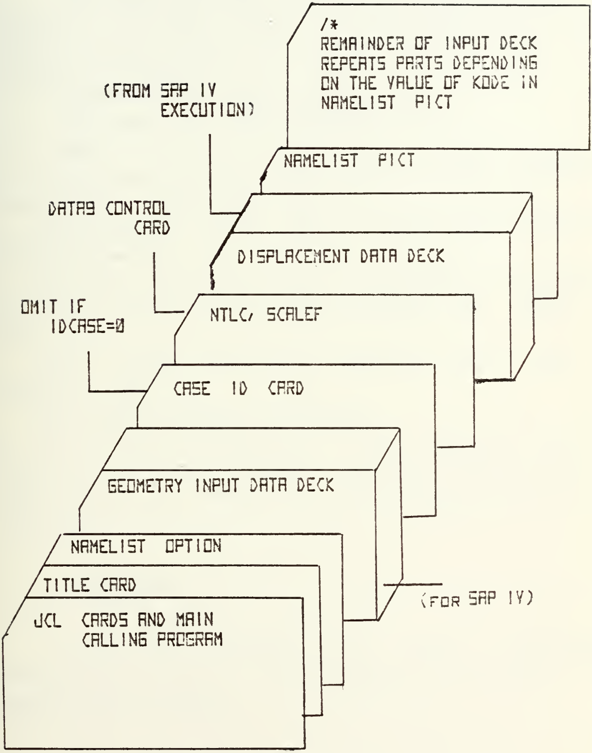


Figure 2 - PSAP INPUT CARD SEQUENCE

3. Main Program

The main program consists of three cards. It allocates the proper storage and calls SUBROUTINE PSAP. An example of a main program for a finite element model that contains 400 nodes is :

```
DIMENSION ZZZ (2800), DISPD (5,3,400)
CALL PSAP (ZZZ, 2800,DISPD,400)
END
```

The definition of the arguments used in calling SUBROUTINE PSAP (ZZZ, NZ, DISPD, NON) are :

ZZZ-blankcommon array used to store nodal coordinates and displacements

NZ-length of array ZZZ (NZ is determined by multiplying the number of nodes in the model by seven, i.e., NZ=no. of nodes * 7)

DISPD-a three dimensional working array used in subroutine DATA9

NON-number of nodes in the model

It is crucial to dimension ZZZ (7*NON) and DISPD (5,3,NON) correctly in the main program. Improper dimensioning of these arrays can cause output errors that are not readily traceable.

4. Title Card

This card contains any desired alphanumeric information in columns 1 to 80. The title will appear at the beginning of the plots.

5. Namelist OPTION

Namelists are a convenient means of inputing the names of several parameters along with their corresponding values. For the NPS IBM 360/67, the format for using namelists is as follows:

1) card 1-&name-beginning in card column 2 where name is the name of the subject namelist
2) succeeding cards- beginning in card column 2, the names of the variables and their values separated by commas
3) final card -&END-starting in card column 2.

An example of a namelist format is shown below:

```
col
2
↓
&OPTION
NNDEST = 400,NUDISP =1,
PSIZE =8.0
&END
```

Any or all parts of a defined namelist may be included, and each parameter may be specified in any order between the &name card and the &END card.

The description of the variable names in Namelist OPTION and their default values are contained in Reference 2. They are given here to assist the user in data preparation.

FORTRAN name - Default value - Description	
NNDEST - 1	The number of nodes(NON) as defined in the program
NUDISP - 1	0 x-direction displacements not to be input 1 x-direction displacements to be input

NVDISP - 1 0 y-direction displacements not to be input
1 y-direction displacements to be input

NWDISP - 1 0 z-direction displacements not to be input
1 z-direction displacements to be input
(NOTE: when SAP IV displacement data is to be used, NUDISP=NVDISP=NWDISP=1; for no displacement data NUDISP=NVDISP=NWDISP=0)

KGEOM - 9 Specifies the subroutine and corresponding method of input for model geometry
1 subroutine GEOM1, a user-supplied subroutine
2 subroutine GEOM2, a user-supplied subroutine
9 subroutine GEOM9, reads in grid points and element data specifically from a SAP IV data deck

KATA - 9 Specifies the subroutine and corresponding method of input for displacement data
1 subroutine DATA1, a user-supplied subroutine
2 subroutine DATA2, a user-supplied subroutine
9 subroutine DATA9, reads a punched output displacement deck from execution of SAP IV

NVALUS - 0 NOT INCORPORATED-ALLOW DEFAULT

IRESEQ - 1 Grid point numbers are stored in the program from 1 to the total number of grid points
0 no internal resequencing of grid points necessary; they are already in ascending order starting with 1
1 resequence grid points from lowest grid point number to highest grid point number

KPLOT - 1 Specifies the type of output device to be used (ALLOW DEFAULT)

XSPACE - 5.0 Space between plots in the y-direction, in inches(see Figure 3 for an explanation of

	axis orientation)
PSIZE - 10.0	Paper size in x-direction, in inches(used in scaling of plots to insure this dimension is not exceeded)
IDCASE - 0	0 no identification card preceeds the deck of displacement values 1 identification card preceeds the deck of displacement values

6. Geometry Input Data Deck

This portion of the input deck contains the grid point locations and the element connectivity. The deck has one of the following forms, depending on the value of KGEOM in the Namelist OPTION.

KGEOM = 9

Calls subroutine GEOM9, which is constructed to read SAP IV geometry data.

(a) When KGEOM is specified as 9, the complete input deck of SAP IV data, prepared according to Appendices I through IV of Reference 1, is placed after the &END card of Namelist OPTION. The portions of SAP IV data deck that involve load, mass, or dynamic analysis data are not part of the input geometry data to SUBROUTINE PSAP. Only the grid point locations and the element connectivity data are used to generate the orthographic projection of the model.

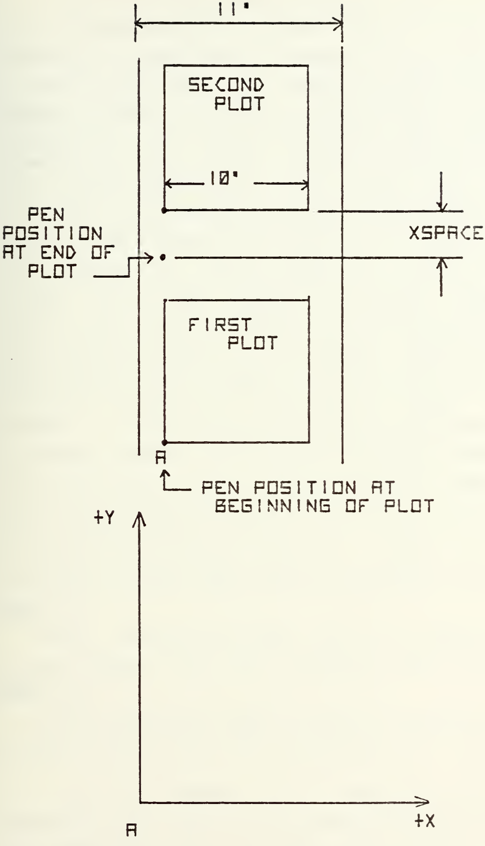


Figure 3 - NPS CALCOMP PLOTTER ORIENTATION

(b) It is possible to input only a portion of the finite element model for a data check. To do so, it is necessary tc modify SAP IV element control cards (described in detail in Reference 1, Appendix IV) for the desired element types to reflect the portion of the element connectivity cards that will be input. For example, if only connectivity for element numbers 15 through 50 of element type 1 (truss elements) of a SAP IV data deck are available for input, it is necessary to alter the element control card for the truss elements. The field on the SAP IV control card that defines the total number of truss elements (card columns 6-10) would reflect the upper bound, in this case-50, and card columns 65-70 would reflect the lower bound, in this case-15. All nodal coordinates for the entire mcdel may be input, or only those that specifically define the portion of the finite element model to be plotted. In either case, the nodal coordinates that relate to element numbers 15-50 must be specified. Unknown results will occur when trying to plot an element whose node points are not specified. The above feature is valuable in a case where several different people are preparing different parts of a large data base for a SAP IV problem and desire to individually check their inputs graphically for accuracy.

<u>KGEOM = 1</u>

Calls subroutine GEOM1, which is prepared by the user to read geometry data from a program other than SAP IV.

<u>KGEOM = 2</u>

Calls subroutine GEOM2, which is prepared by the user to read geometry data from a program other than SAP IV.

Use of KGEOM=1 or 2 requires modification of SUBROUTINE PSAP to fit the specific format of the user's input gecmetry data. A method for doing this will be discussed in paragarph C. of this section.

7. Case Indentification Card

If IDCASE =0 is specified in Namelist OPTION, this card is omitted. The card, if present, contains any desired alphanumeric information in card columns 1-80 which will identify all displacements for a given case. For IDCASE = 1 and SAP IV punched displacement data, a case identification card must appear before each Namelist PICT for every different load case that is plotted in addition to load case one. This is illustrated in Figure 4. A maximum number of five different load cases can be obtained from SAP IV. The case ID card information, if present, will appear before each load case's DISPLACEMENT DATA TO BE PLOTTED section in the printed computer output from SUBROUTINE PSAP. This information does not appear on any Calcomp plots.

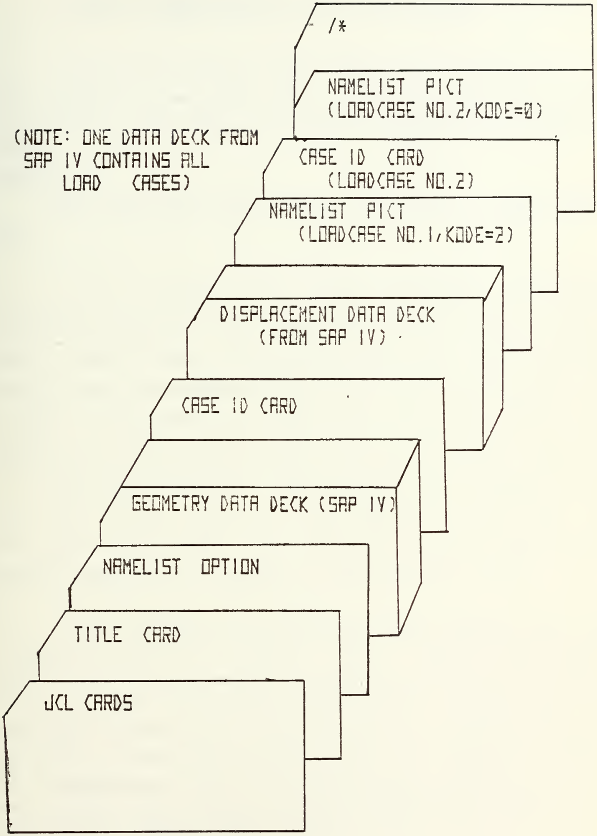


Figure 4 - SAMPLE SUBROUTINE PSAP EXECUTION DECK WITH IDCASE=1

8. <u>Deck of Displacement Data to be Plotted</u>

This deck contains the displacement values of the nodal points. The deck has one of the following forms, depending on the value of KDATA specified in Namelist OPTION.

<u>KDATA = 9</u>

Calls subroutine DATA9, which reads SAP IV displacement data.

(a) A single card, format (I5,F10.0), containing the number of total load cases (NTLC) that are in the SAP IV output displacement data deck, and a scale factor (SCALEF) that is used in scaling the displacement data, must be input before the displacement deck. This card controls the input of SAP IV displacement data through subroutine DATA9. For example, a displacement deck from an execution of SAP IV that contains the maximum of five load cases would have NTLC = 5. The scale factor could be any value desired by the user, with default equal to 1. The DATA9 control card for the above case would be a 5 in card column five and the desired scale factor (SCALEF) in card columns 6-15.

(b) The deck of displacement data obtained from SAP IV follows the DATA 9 control card. Since the maximum number of load cases that are punched out by SAP IV is five, the maximum NTLC is five and the actual number must be specified. Note that if NUDISP, NVDISP, and NWDISP are all specified as zeroes through default or in Namelist OPTION, the displacement data deck and the DATA 9 control card are not required. This feature enables preprocessing only of a given finite element model by PSAP.

(c) The parameter DMAGS in Namelist PICT also provides for magnification of displacements. (See section III.B.9)

KDATA = 1

Calls subroutine DATA1, which is prepared by the user to read displacement data from a program other than SAP IV to be plotted.

KDATA = 5

Calls subroutine DATA 5, which is prepared by the user to read displacement data from a program other than SAP IV to be plotted.

Use of KDATA =1 or 5 requires that SUBROUTINE PSAP be modified to fit the format that the user's input displacement data specifically requires. A method for accomplishing this change will be discussed in parapgraph C. of this section.

9. Namelist PICT

The format for Namelist PICT is the same as that required for Namelist OPTION. It requires a single card, &PICT, followed by the specified parameter cards and the &END cards, all cards beginning in card column 2. This namelist contains the values needed to specify the type of plot that is desired and what information is to be included on the plots. A detailed summary of Namelist PICT is contained in Reference 2 and is given here for user convenience.

FORTRAN name - Default value - Description

KHORZ - 1 Integer designating the horizontal axes of the viewing plane where $1=X_0$, $2=Y_0$, and $3=Z_0$. (see Figure 5.)

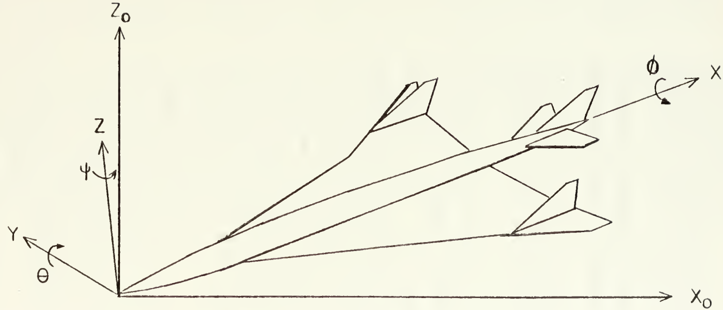


Figure 5 - COORDINATE SYSTEM AND ROTATIONS FOR AN OBLIQUE ORTHOGRAPHIC PROJ. SHOWN IN X - Z VIEWING PLANE

KVERT - 2	Integer designating the vertical axes of the viewing plane where $1=X_0$, $2=Y_0$, and $3=Z_0$. (see Figure 5.)
PHI - 0.0	Angular rotation of model about its X-axis in degrees (must be performed third, see Figure 5.)
THETA - 0.0	Angular rotation of model about its Y-axis in degrees (must be performed second, see Figure 5.)
PSI - 0.0	Angular rotation of model about its Z-axis in degrees (must be performed first, see Figure 5.)
NEWFR - 1	1 Frame change before plotting (a frame change resets the Y-origin past previous plot by XSPACE given in Namelist OPTION and resets the X-origin at 0.0) 0 no frame change before plotting
ISCALE - 1	1 automatic scaling of plot and computation of proper origin location 2 user-specified origin and scaling
PLOTSZ - 10.0	Maximum dimension desired on completed plot, in inches (used for scaling if ISCALE=1)
XORGN - 0.0	X-location of plot origin (used if ISCALE=2)
YORGN - 0.0	Y-location of plot origin (used if ISCALE=2)
PSCALE - 1.0	Model size reduction factor (i.e., PSCALE is equal to actual model size divided by desired plot size, used if ISCALE=2)
NOTAT - 0	0 no numbering on plots 1 numbering of grid points 2 numbering of elements
XLHT - 0.15	Height of integers specified by NOTAT, in inches (must be ≥ 0.07)
KDISP - 0	0 plot of undeformed structure

	1 plot of deformed structure
	2 exploded plot
	3 displacements represented by vectors
IDMAG - 2	1 direct magnification of displacement data by DMAGS
	2 scaling of displacement data to a maximum value ofDMAGS
DMAGS - 1.0	Magnification of displacements (if KDISP= 1 or 3)
	Reduction factor of elements (if KDISP=2)
KSYMXY - 0	1 symmetry about X-Y plane
KSYMXZ - 0	1 symmetry about X-Z plane
KSYMYZ - 0	1 symmetry about Y-Z plane

Symmetries are performed consecutively (i.e., a plate quadrant with KSYMXZ and KSYMYZ equal to one would yield a complete plate).

XXMAX,XXMIN - 1.0E+20	Locate cutting planes parallel to principal planes
YYMAX,YYMIN - 1.0E-20	(X-Y, X-Z, Y-Z) to limit plot
ZZMAX,ZZMIN	

NDMAX - 9999999	Maximum grid point identification number to be included in the plot
NDMIN - 0	Minimum grid point identification number to be included in the plot
NELMAX - 9999999	Maximum element identification number to be included in the plot
NELMIN - 0	Minimum element identification number to be included in the plot
KODE - 0	Specifies control option after a plot is complete 0 last plot, exit from program 1 read another Namelist PICT

2 read a new set of displacement data (see NOTE 1 below)

3 read a complete new set of input data starting with a title card.

(NOTE 1: For SAP IV displacement data, KODE=2 signifies that the next load case displacements will be assigned to the model nodal points.)

The previous sections describe a complete basic set of input data, if KODE =0 in Namelist PICT. For KODE =1, 2, or 3, additional sections of the deck must be repeated. The deck must end with a Namelist PICT having a value of KODE = 0 in it. An example input data deck and output plots for the simple truss problem of Appendix A is found in Appendix B.

C. METHOD FOR ALTERING SUBROUTINE PSAP

In the event a user has geometry and displacement data decks from a program other than SAP IV, it is possible to plot those decks with PSAP. The subroutine will handle rod-like elements, triangular elements, or quadrilateral elements when they are input in acceptable format. By studying subroutines GEOM9 and DATA9, and SUBROUTINE PSAP (Appendix D), the necessary sequence of input can be determined. PSAP is presently constructed so that a user may supply his own routines through the use of subroutines GEOM1 or GEOM2 for geometry data and DATA1 or DATA5 for displacement data. In order to add a subroutine to PSAP, it need only be placed after the main calling program in the sequence of control cards as discussed in paragraph II.B.2. The essential features for the input of the geometry data are the nodal points, with their X,Y,Z coordinates, and the connectivity sequence for the finite element model. The

necessary part of displacement data input is the node point number, with the U,V,W displacements. Adding a user-prepared subroutine through GEOM1 or 2 and DATA1 or 5 should prove to be a relatively straight-forward task for the user who desires to do so. The listing of GEOM9 in Appendix D can be used as a guide.

D. SIGNIFICANT ASPECTS OF SUBROUTINE PSAP

1. Exploded Plots

Often the absence or presence of elements in a finite element model cannot be determined from a conventional oblique orthographic projection. For example, a line element that is coincident with an edge of a triangular or quadrilateral element could not be detected. To show clearly each element, PSAP contains an algorithm for generating exploded oblique orthographic projections. This can be a valuable tool in checking the toplogy of an analytical model. (i.e., KDISP = 2)

2. Portions of Models

The ability to isolate a portion of a model for detailed examination is a very useful and desirable asset of a preprocessor. SUBROUTINE PSAP has the capability of specifying cutting planes (i.e.,XXMIN ,XXMAX ,YYMIN,YYMAX, etc.) or maximum and minumum element numbers (i.e., NELMIN, NELMAX). Examples of this are shown in Appendix D.

3. Specification of View

The specification of view of a model is done through the use of the parameters KVERT,KHORZ, PSI, THETA, and PHI that are found in Namelist PICT (Section III.B.9) The specific details of how this is accomplished can be found in Reference 2. There are a great number of possible combinations of the above parameters, and Appendix C illustrates various combinations that were used on a section of a finite element wing model. The Calcomp plots, along with the parameters as specified, are found in each figure.

4. Possible Sources of Errors

(a) The most probable source of error could be incorrect data deck preparation or deck sequencing. It is important to use Figure 2 as a guide while preparing the input for PSAP.

(6) Other errors may occur if the arrays ZZZ and DISPD are not dimensioned correctly as discussed in paragraph III.B.3. During the execution of the program several manipulations are performed with the two arrays, and it is possible for addresses of the data to be lost within the IBM 360/67. The 360 System error messages may not directly indicate that the dimensions of the arrays have been exceeded.

(c) A most important point to remember in generating a sequence of plots is that once a parameter has been assigned a value in a namelist, it retains that value until it is reassigned. For example, if PLOTSZ is assigned a value of 8.0 for the first in a series of plots and it is not

redefined in any subsequent Namelists PICT , the value of PLOTSZ will retain the value (8.0), as originally specified.

(d) It is possible to make any number of errors, however, and all of them cannot be anticipated. The error messages from the IBM 360/67 System will, in most cases, be straight-forward and facilitate easy trouble-shooting.

IV. CONCLUSIONS AND RECOMMENDATIONS

General purpose finite element structural analysis programs like SAP IV are significant analytical tools that can be used most effectively when coupled with a flexible pre- and postprocessor. The availability of a powerful structural analysis tool with partial pre- and postprocessing capability has now been provided for Naval Postgraduate School Students. However, at present this capability is limited to pre- and postprocessing the data of only five of the SAP IV elements. It does not appear feasible, at this time, to add any more capability as far as the number of SAP IV elements goes. However, other additional improvements to postprocessing at NPS should be made. For example, Appendix C of Reference 2 lists a program developed to produce contour plots of stress data from finite element models. The program, in all probability, could be adapted to SAP IV and the NPS IBM 360/67. This represents a possible avenue to further the present structural analysis capability for NPS students. SAP IV itself can be expanded. There is a great deal of work being done in the area of composite materials, and SAP IV possesses only very limited orthotropic material capability. However, there are subroutines within SAP IV that could be modified to reflect more current methods in handling ccmposite materials.

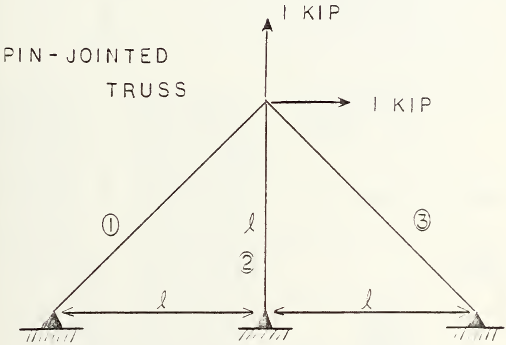
In conclusion, the finite element method in structural engineering is the most powerful analysis tool available today and needs to be exploited by NPS students in both classroom and research work. Programs like SAP IV are available at many Navy laboratories, development centers,

and aerospace companies throughout the country. The current trends in structural analysis are toward computer-aided techniques, and exposure to a general purpose package for students enrolled in AE 4101 and AE 4102 is a significant asset.

APPENDIX A

EXAMPLE PROBLEM USING SAP IV

The schematic given below illustrates the example problem for which the data on the following page were prepared.



l = 100 inches

E = 10.1E+06 psi

ALPHA = 12.6E-06 psi

A = 0.5 sq.in.

A = 1.0 sq.in.

A = 0.75 sq.in.

```
// (STANDARD GREEN JOB CARD)
//GO EXEC PGM=SAP,REGION=260K
//STEPLIB DD UNIT=2321,VOL=SER=CEL002,DISP=SHR,DSN=F0559.SAPLM
//GO.FT01F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT02F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT03F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT04F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT05F001 DD DDNAME=SYSIN
//GO.FT06F001 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330),
//    SPACE=(CYL,(3,1))
//GO.FT07F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT08F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT09F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT10F001 DD UNIT=SYSDA,SPACE=(CYL,(6,6)),
//    DCB=(RECFM=VSB,LRECL=1284,BLKSIZE=2572)
//GO.FT11F001 DD SYSOUT=B,SPACE=(TRK,(20,2)),
//    DCB=(RECFM=FB,LRECL=80,BLKSIZE=7200)
//GO.SYSIN DD *
  SAP IV  -  EXAMPLE(INPUT DATA)
    4    1    1    0    0    0    0    0
    1    1    1    1    1    1    1       0.0          0.0          0.0
    2    1    1    1    1    1    1     100.          0.0         0.0
    3    1    1    1    1    1    1     200.          0.0         0.0
    4    0    0    1    1    1    1     100.0       100.0          0.0
    1    3    3
    110100000.0  .0000126        0       0.5          0.1
    210100000.0  .0000126        0       1.0          0.1
    310100000.0  .0000126        0      0.75          0.1
(BLANK CARD)
(BLANK CARD)
(BLANK CARD)
(BLANK CARD)
    1    1    4    1         0.0     0
    2    2    4    2         0.0     0
    3    3    4    3         0.0     0
    4    1  1000.      1000.
(BLANK CARD)
(BLANK CARD)
(BLANK CARD)
/ ( STANDARD NPS EOF CARD )
```

APPENDIX B

EXAMPLE PROBLEM USING SUBROUTINE PSAP

The pin-jointed truss problem in Appendix A is also used as an example in illustrating SUBROUTINE PSAP use. The entire computer card deck used to generate Figures 6, 7, and 8 is listed on the following two pages of this appendix. The printed computer output for this problem is shown in Figure 9. Note that the displacements of node number 4 have been magnified by 1000.

```
// (STANDARD GREEN JOB CARD)
//  EXEC FORTCLGP,REGION.GO=180K
//FORT.SYSIN DD *
       DIMENSION ZZZ(100),DISPD(5,3,4)
       CALL PSAP(ZZZ,100,DISPD,4)
       END
/ ( STANDARC NPS EOF CARD )
//LINK.USDD DD UNIT=2321,VOL=SER=CEL002,DISP=SHR,DSN=F0559.PSAPLM
//LINK.SYSIN DD *
      INCLUDE USDD(PSAP)
      ENTRY MAIN
/ ( STANCARC NPS EOF CARD )
//GO.FT10F001 DD UNIT=SYSDA,
// SPACE=(CYL,(3,1)),
// DCB=(RECFM=VS,BLKSIZE=3520)
//GO.SYSIN DD *
    PIN-JOINTED TRUSS ----PSAP EXAMPLE---
 &OPTION
 NNCEST=4,NUDISP=1,NVDISP=1,NWDISP=1
 &END
  SAP IV  -  EXAMPLE(INPUT DATA)
    4    1    1    0    0    0    0    0
    1    1    1    1    1    1    1        0.0          0.0          0.0
    2    1    1    1    1    1    1      100.           0.0         0.0
    3    1    1    1    1    1    1      200.           0.0         0.0
    4    0    0    1    1    1    1      100.0        100.0          0.0
    1    3    3
    110100000.0  .0000126       0         0.5          0.1
    210100000.0  .0000126       0         1.0          0.1
    310100000.0  .0000126       0        0.75          0.1
(BLANK CARD)
(BLANK CARD)
(BLANK CARD)
(BLANK CARD)
    1    1    4    1       0.0    0
    2    2    4    2       0.0    0
    3    3    4    3       0.0    0
    1    1000.
   4    1 0.24072E-01 0.83420E-02 0.0          0.0          0.0          0.0
   3    1 0.0         0.0         0.0          0.0          0.0          0.0
   2    1 0.0         0.0         0.0          0.0          0.0          0.0
   1    1 0.0         0.0         0.0          0.0          0.0          0.0
 &PICT
 PLOTSZ=5.0,NOTAT=1,KODE=1
 &END
 &PICT
 NOTAT=2,KDISP=1,KODE=1,IDMAG=1
 &END
```

```
 &PICT
 NOTAT=1,KDISP=3,KODE=0
 &END
/ ( STANDARD NPS EOF CARD )
```

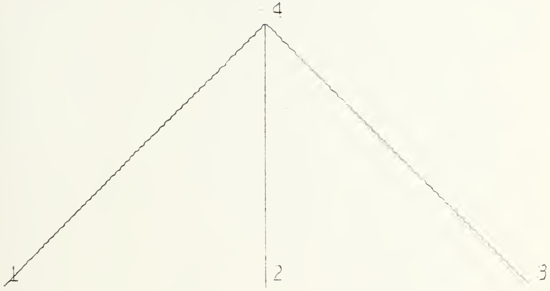


Figure 6 - UNDEFORMED TRUSS MODEL

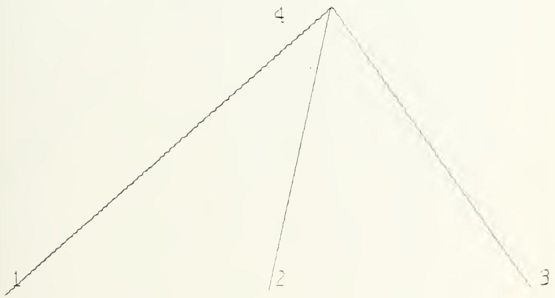


Figure 7 - DEFORMED TRUSS MODEL(NODE4 DISPLACED)

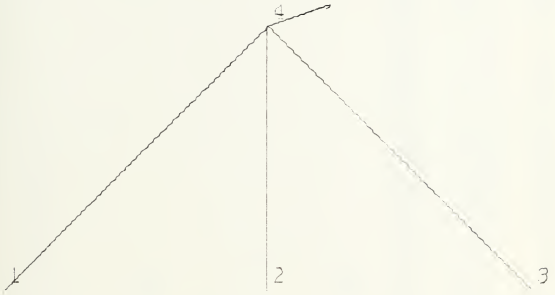


Figure 8 - DISPLACEMENTS(NODE4) FOR TRUSS MODEL PLOTTED AS A VECTOR

```
                    PIN-JOINTED TRUSS ----PSAP EXAMPLE---


                 BLANK COMMON STORAGE ZZZ REQUIRES AT LEAST      28 LOCATIONS FOR THIS CASE


      GRID POINT INFORMATION

      RESEQUENCED     USER INPUT
      GRID POINT      GRID POINT
      NUMBER          NUMBER             X              Y              Z

           1               1        0.0            0.0            0.0
           2               2        0.1000E 03     0.0            0.0
           3               3        0.2000E 03     0.0            0.0
           4               4        0.1000E 03     0.1000E 03     0.0

      ELEMENT INFORMATION - WITH RESEQUENCED GRID POINTS

      RESEQUENCED     USER INPUT                    GRID POINTS
      ELEMENT         ELEMENT
      NUMBER          NUMBER             1          2          3          4

           1               1             1          4          0          0
           2               2             2          4          0          0
           3               3             3          4          0          0

      DISPLACEMENTS TO BE PLOTTED

      RESEQUENCED     USER INPUT
      GRID POINT      GRID POINT
      NUMBER          NUMBER             X              Y              Z

           1               1        0.0            0.0            0.0
           2               2        0.0            0.0            0.0
           3               3        0.0            0.0            0.0
           4               4        0.2407E 02     0.8342E 01     0.0

&PICT
KHORZ=           1,KVERT=           2,PHI= 0.0          ,THETA= 0.0          ,PSI= 0.0          ,NEWFR=           1,ISCALE=
         1,PLOTSZ=   5.0000000   ,XORGN= 0.0          ,YORGN= 0.0          ,PSCALE=  1.0000000   ,NOTAT=           1,KOISP=
         0,IDMAG=           2,DMAGS=  1.0000000   ,KODE=           1,KSYMXY=           0,KSYMXZ=           0,KSYMYZ=           0,
XXMAX= 0.99999984E 20,YYMAX= 0.99999984E 20,ZZMAX= 0.99999984E 20,XXMIN=-0.99999984E 20,YYMIN=-0.99999984E 20,ZZMIN=-0.99999984E 20,
NOMAX=     9999999,NOMIN=           0,NELMAX=     9999999,NELMIN=           0,XLHT= 0.14999998
&END

&PICT
KHORZ=           1,KVERT=           2,PHI= 0.0          ,THETA= 0.0          ,PSI= 0.0          ,NEWFR=           1,ISCALE=
         1,PLOTSZ=   5.0000000   ,XORGN= 0.0          ,YORGN= 0.0          ,PSCALE=  40.000000   ,NOTAT=           1,KOISP=
         1,IDMAG=           1,DMAGS=  1.0000000   ,KODE=           1,KSYMXY=           0,KSYMXZ=           0,KSYMYZ=           0,
XXMAX= 0.99999984E 20,YYMAX= 0.99999984E 20,ZZMAX= 0.99999984E 20,XXMIN=-0.99999984E 20,YYMIN=-0.99999984E 20,ZZMIN=-0.99999984E 20,
NOMAX=     9999999,NOMIN=           0,NELMAX=     9999999,NELMIN=           0,XLHT= 0.14999998
&END

&PICT
KHORZ=           1,KVERT=           2,PHI= 0.0          ,THETA= 0.0          ,PSI= 0.0          ,NEWFR=           1,ISCALE=
         1,PLOTSZ=   5.0000000   ,XORGN= 0.0          ,YORGN= 0.0          ,PSCALE=  40.000000   ,NOTAT=           1,KOISP=
         3,IDMAG=           1,DMAGS=  1.0000000   ,KODE=           0,KSYMXY=           0,KSYMXZ=           0,KSYMYZ=           0,
XXMAX= 0.99999984E 20,YYMAX= 0.99999984E 20,ZZMAX= 0.99999984E 20,XXMIN=-0.99999984E 20,YYMIN=-0.99999984E 20,ZZMIN=-0.99999984E 20,
NOMAX=     9999999,NOMIN=           0,NELMAX=     9999999,NELMIN=           0,XLHT= 0.14999998
&END
```

Figure 9 - SAMPLE COMPUTER PRINTOUT FROM PSAP OF TRUSS MODEL

APPENDIX C

SAMPLE SUBROUTINE PSAP OUTPUTS FOR A FINITE ELEMENT MODEL OF A WING

The figures contained in this appendix are of a finite element wing model developed in Flight Vehicle Structural Analysis II (AE 4102, QTR.IV, 1976). The original wing was designed in AE 4274, which is the latter half of the subsonic structural design sequence in the Department of Aeronautics. The finite element model of the wing was developed in AE 4102 as a class project. (The structural analysis and design sequences in the the Department of Aeronautics are constructed so that they interelate the two different areas.) Each student was assigned responsibility for a portion of the wing and the student whose design was used was the analysis leader.

Figure 10 shows the axis orientation used in the original definition of the wing as designed in AE 4274. Figure 11 and 12 show the entire model with over four hundred membrane elements and with different views of the model specified. In Figures 13 and 14 a single portion of the wing is shown with different rotation angles. The significant Namelist PICT values used to orient the model are included in the figures.

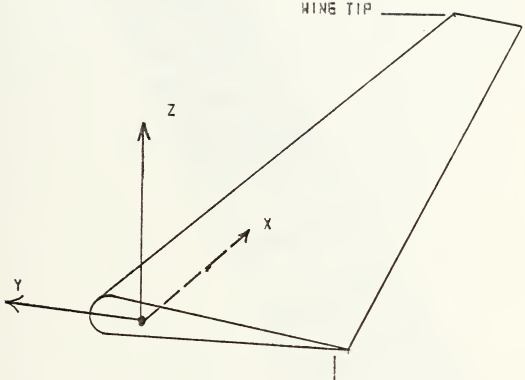


Figure 10 - WING MODEL AXIS ORIENTATION

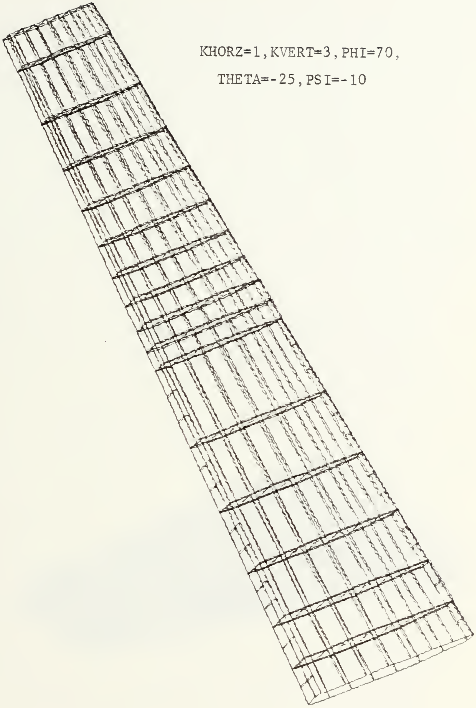


Figure 11 - ORTHOGRAPHIC PROJECTION OF WING MODEL -1

KHORZ=2,KVERT=3,PHI=60,
THETA=-25,PSI=-10

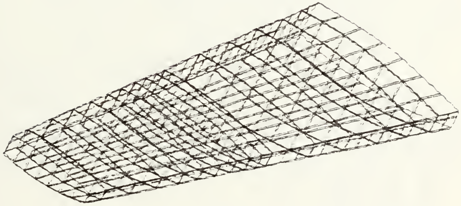
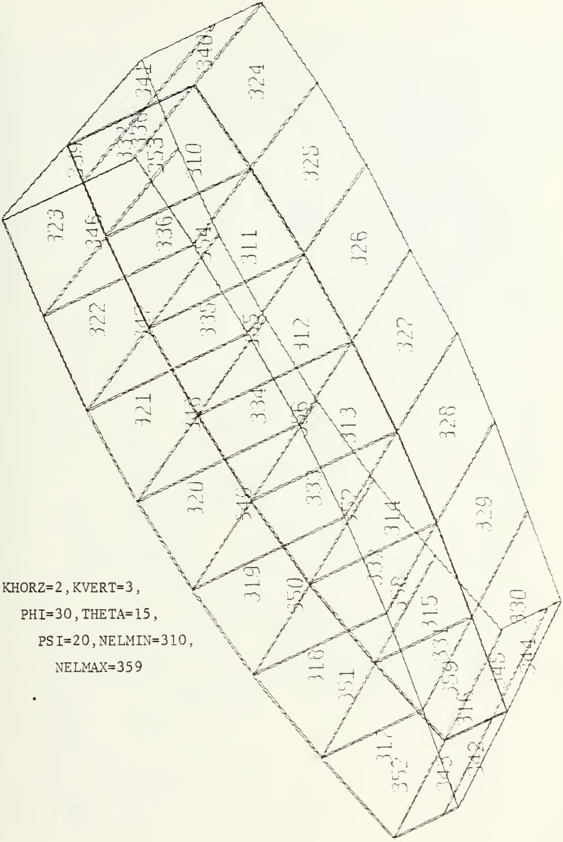


Figure 12 - ORTHOGRAPHIC PROJECTION OF WING MODEL -2



KHORZ=2,KVERT=3,
PHI=30,THETA=15,
PSI=20,NELMIN=310,
NELMAX=359

Figure 13 - WING MODEL SECTION - ORIENTATION -1

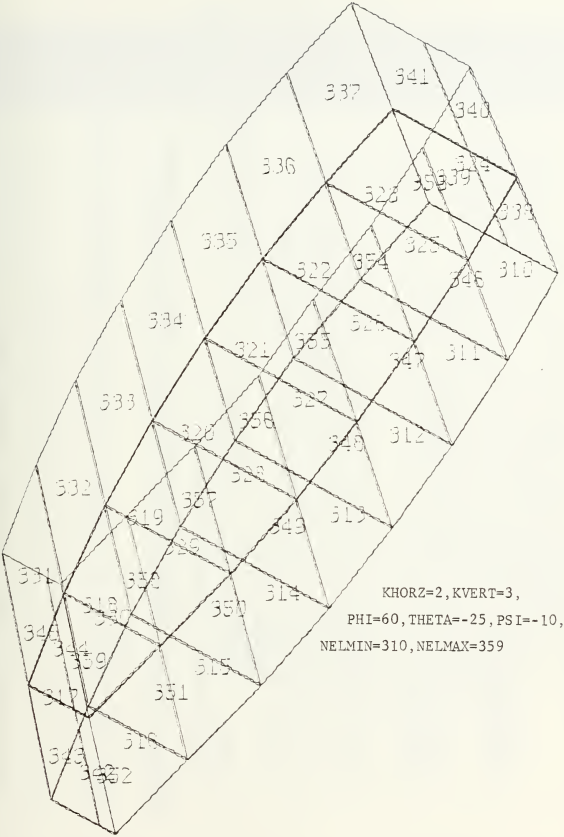


Figure 14 - WING MODEL SECTION - ORIENTATION -2

APPENDIX D

SUBROUTINE PSAP LISTING

```
C
C *** THIS SECTION CONTAINS SUBROUTINE DOCUMENTATION
C
C
C
C                      DESCRIPTION OF INPUT DATA CARDS
C
C
C
C    TITLE CARD - CONTAINS ANY DESIRED ALPHANUMERIC INFORMATION IN COLS.
C                 1-80.
C
C
C    NAMELIST OPTION - CONTAINS VALUES TO VERIFY STORAGE IN BLANK
C                      COMMON AND CONTROL VALUES NEEDED BY THE PROGRAM.
C
C       THE FOLLOWING VALUES ARE INCLUDED---
C
C       NNDEST = ESTIMATE NUMBER OF GRID POINTS TO BE USED.  VALUE MUST
C                BE GREATER THAN OR EQUAL TO THE ACTUAL NUMBER OF GRID
C               POINTS.
C           ** DEFAULT = 200 **
C       NUDISP = 0 FOR NO DISPLACEMENT DATA IN X-DIRECTION.
C              = 1 FOR DATA INCLUDING DISPLACEMENTS IN X-DIRECTION.
C           ** DEFAULT = 1 **
C       NVDISP = 0 FOR NO DISPLACEMENT DATA IN Y-DIRECTION.
C              = 1 FOR DATA INCLUDING DISPLACEMENTS IN Y-DIRECTION.
C           ** DEFAULT = 1 **
C       NWDISP = 0 FOR NO DISPLACEMENT DATA IN Z-DIRECTION.
C              = 1 FOR DATA INCLUDING DISPLACEMENTS IN Z-DIRECTION.
C           ** DEFAULT = 1 **
C
C       KGEOM SPECIFIES SUBROUTINE AND CORRESPONDING METHOD OF INPUT
C       FOR MODEL GEOMETRY.
C       KGEOM = 1 FOR USER SUPPLIED SUBROUTINE - GEOM1
C             = 2 FOR USER SUPPLIED SUBROUTINE - GEOM2
C             = 9 FOR SAP IV DATA DECK INPUT SUBROUTINE - GEOM9.
C           ** DEFAULT = 9 **
C       KDATA SPECIFIES SUBROUTINE AND CORRESPONDING METHOD OF INPUT
C       FOR DISPLACEMENT DATA.
C       KDATA = 1 FOR SUBROUTINE DATA1 TO READ IN DISPLACEMENT DATA
C                -- SUPPLIED BY THE USER.
C             = 5 FOR SUBROUTINE DATA5 TO READ IN DISPLACEMENT DATA
C                 -- SUPPLIED BY THE USER.
C             = 9 FOR SUBROUTINE DATA9 TO READ SAP IV DATA.
C           ** DEFAULT = 9 **
C      NVALUS - NOT USED AT NPS ----- ALLOW DEFAULT
C
```

```
C               ** DEFAULT = 0 **                                       DOC00490
C        IRESEQ = 0 FOR NO RESEQUENCING OF GRID PCINT NUMBERS.          DOC00500
C               = 1 TO RESEQUENCE GRID POINT NUMBERS IN SAME ORDER      DOC00510
C                   AS THEY ARE INPUT.                                  DOC00520
C               ** DEFAULT = 1 **                                       DOC00530
C        KPLOT SPECIFIES THE TYPE OF OUTPUT DEVICE TO BE USED.          DOC00540
C        KPLOT = 1 FOR CALCOMP.                                         DOC00550
C              = 2 FOR LANGLEY RESEARCH CENTER USE ONLY                 DOC00560
C              = 3 FOR LRC USE ONLY.                                    DOC00570
C              = 4 FOR   LRC USE ONLY                                   DOC00580
C               ** DEFAULT = 1 **                                       DOC00590
C        XSPACE = SPACE BETWEEN PLOTS IN Y-DIRECTION, IN INCHES.        DOC00600
C               ** DEFAULT = 5.0 **                                     DOC00610
C        PSIZE = PAPER SIZE IN Y-DIRECTION IN INCHES, USED IN SCALING OFDOC00620
C                 PLOTS TO INSURE THIS DIMENSION IS NOT EXCEEDED.       DOC00630
C               ** DEFAULT = 10.0 **                                    DOC00640
C        IDCASE = 0 FOR NO TITLE CARD PRECEDING                         DOC00650
C                   DECKS OF DISPLACEMENT VALUES.                       DOC00660
C               = 1 FOR TITLE CARD PRECEDING                            DOC00670
C                   DECKS OF DISPLACEMENT VALUES.                       DOC00680
C               ** DEFAULT = 0 **                                       DOC00690
                                                                        DOC00700
                                                                        DOC00710
                                                                        DOC00720
C     MODEL GEOMETRY IS NOW INPUT IN ONE OF THE FOLLOWING FORMS,        DOC00730
C        DEPENDING ON THE VALUE OF KGEOM SPECIFIED IN NAMELIST OPTION.  DOC00740
C                                                                       DOC00750
C                                                                       DOC00760
C       USE IF KGEOM = 1                                                DOC00770
C         CALL SUBROUTINE GEOM1 WHICH IS PREPARED BY THE USER TO        DOC00780
C         READ GEOMETRY DATA.                                           DOC00790
C                                                                       DOC00800
C       USE IF KGEOM = 2                                                DOC00810
C         CALL SUBROUTINE GEOM2 WHICH IS PREPARED BY THE USER TO        DOC00820
C         READ GEOMETRY DATA.                                           DOC00830
C                                                                       DOC00840
C                                                                       DOC00850
C                                                                       DOC00860
C       USE IF KGEOM = 9                                                DOC00870
C                                                                       DOC00880
C          CALL SUBROUTINE GEOM9 WHICH READS SAP IV GEOMETRY DATA.      DOC00890
C                                                                       DOC00900
C                                                                       DOC00910
C                                                                       DOC00920
C     CASE IDENTIFICATION CARD.                                         DOC00930
C                                                                       DOC00940
C          THIS CARD IS OMITTED IF IDCASE=0 IS SPECIFIED IN $OPTION.    DOC00950
C          IF PRESENT, THIS CARD CONTAINS ANY DESIRED ALPHANUMERIC      DOC00960
```

```
C        INFORMATION IN COLS.1-80.  WILL APPEAR BEFORE EACH DATA PLOT.  DOC00970
C                                                                       DOC00980
C                                                                       DOC00990
C    CATA TO BE PLOTTED IS NOW INPUT IN ONE OF THE FOLLOWING FORMS,     DOC01000
C        DEPENDING ON THE VALUE OF KDATA SPECIFIEC IN NAMELIST OPTION.  DOC01010
C                                                                       DOC01020
C                                                                       DOC01030
C      USE IF KDATA = 1                                                 DOC01040
C       CALL SUBROUTINE DATA1 WHICH IS PREPARED BY THE USER             DOC01050
C                                                                       DOC01060
C                                                                       DOC01070
C                                                                       DOC01080
C      USE IF KDATA = 5                                                 DOC01090
C       CALL SUBROUTINE DATA5 WHICH IS PREPARED BY THE USER             DOC01100
C                                                                       DOC01110
C                                                                       DOC01120
C      USE IF KDATA = 9                                                 DOC01130
C                                                                       DOC01140
C        CALL SUBROUTINE DATA9 WHICH READS SAP IV DISPLACEMENT CATA.    DOC01150
C                                                                       DOC01160
C                                                                       DOC01170
C                                                                       DOC01180
C                                                                       DOC01190
C     NAMELIST PICT - CONTAINS VALUES NEEDED TO GENERATE PLOTS.         DOC01200
C                                                                       DOC01210
C        THE FOLLOWING VALUES ARE INCLUCED---                           DOC01220
C                                                                       DOC01230
C        KHORZ = INTEGER DESIGNATING HORIZONTAL AXIS OF VIEWING PLANE,  DOC01240
C                 WHERE 1=X, 2=Y, 3=Z.                                  DOC01250
C             ** DEFAULT = 1 **                                         DOC01260
C        KVERT = INTEGER DESIGNATING VERTICAL AXIS OF VIEWING PLANE,    DOC01270
C                 WHERE 1=X, 2=Y, 3=Z.                                  DOC01280
C             ** DEFAULT = 2 **                                         DOC01290
C        PHI = ANGULAR ROTATION OF MODEL ABOUT ITS X-AXIS, IN DEGREES   DOC01300
C               (MUST BE TAKEN THIRD).                                  DOC01310
C             ** DEFAULT = 0.0 **                                       DOC01320
C        THETA = ANGULAR ROTATION OF MODEL ABOUT ITS Y-AXIS, IN DEGREES DOC01330
C                (MUST BE TAKEN SECOND).                                DOC01340
C             ** DEFAULT = 0.0 **                                       DOC01350
C        PSI = ANGULAR ROTATION OF MODEL ABOUT ITS Z-AXIS, IN CEGREES   DOC01360
C               (MUST BE TAKEN FIRST).                                  DOC01370
C             ** DEFAULT = 0.0 **                                       DOC01380
C        NEWFR = 1 FOR FRAME CHANGE BEFORE PLOT IS MADE.                DOC01390
C                  (A FRAME CHANGE RESETS THE Y-ORGIN PAST PREVIOUS PLOTDOC01400
C                   BY XSPACE AND THE X-ORGIN AT 0.0).                  DOC01410
C        NEWFR .NE. 1 FOR NO FRAME CHANGE BEFORE PLOTTING.              DOC01420
C             ** DEFAULT = 1 **                                         DOC01430
C        ISCALE = 1 FOR INTERNAL ORGIN LOCATION AND SCALING.            DOC01440
```

```
C              = 2 FOR USER SPECIFIED ORGIN AND SCALING.                      DOC01450
C           ** DEFAULT = 1 **                                                 DOC01460
C       PLOTSZ = MAXIMUM DIMENSION DESIRED ON COMPLETED PLOT.                 DOC01470
C                (USED FOR SCALING IF ISCALE = 1)                             DOC01480
C           ** DEFAULT = 10.0 **                                              DOC01490
C       XORGN = X-LOCATION OF PLOT ORIGIN (USED IF ISCALE = 2).               DOC01500
C           ** DEFAULT = 0.0 **                                               DOC01510
C       YORGN = Y-LOCATION OF PLOT ORIGIN (USED IF ISCALE = 2).               DOC01520
C           ** DEFAULT = 0.0 **                                               DOC01530
C       PSCALE = MODEL SIZE REDUCTION FACTOR, PSCALE = ACTUAL MODEL           DOC01540
C                SIZE/DESIRED PLOT SIZE (USED IF ISCALE = 2).                 DOC01550
C           ** DEFAULT = 1.0 **                                               DOC01560
C       NOTAT = 0 FOR NO NUMBERING ON PLOTS.                                  DOC01570
C             = 1 FOR NUMBERING OF GRID POINTS.                               DOC01580
C             = 2 FOR NUMBERING OF ELEMENTS.                                  DOC01590
C           ** DEFAULT = 0 **                                                 DOC01600
C       XLHT = HEIGHT OF INTEGERS SPECIFIED BY NOTAT, IN INCHES.              DOC01610
C           ** DEFAULT = 0.15 **                                              DOC01620
C       KDISP = 0 FOR UNDEFORMED PLOT.                                        DOC01630
C             = 1 FOR DEFORMED PLOT.                                          DOC01640
C             = 2 FOR EXPLODED PLOT.                                          DOC01650
C             = 3 FOR DISPLACEMENTS REPRESENTED BY VECTORS.                   DOC01660
C           ** DEFAULT = 0 **                                                 DOC01670
C       IDMAG = 1 FOR DIRECT SCALING OF DATA BY DMAGS.                        DOC01680
C             = 2 FOR SCALING OF DATA TC A MAX. VALUE OF DMAGS.               DOC01690
C           ** DEFAULT = 2 **                                                 DOC01700
C       DMAGS = MAGNIFICATION OF DISPLACEMENTS (IF KDISP=1).                  DOC01710
C             = REDUCTION FACTOR OF ELEMENTS (IF KDISP=2).                    DOC01720
C           ** DEFAULT = 1.0 **                                               DOC01730
C       KSYMXY = 1 FOR SYMMETRY ABOUT X-Y PLANE.                              DOC01740
C           ** DEFAULT = 0 **                                                 DOC01750
C       KSYMXZ = 1 FOR SYMMETRY ABOUT X-Z PLANE.                              DOC01760
C           ** DEFAULT = 0 **                                                 DOC01770
C       KSYMYZ = 1 FOR SYMMETRY ABOUT Y-Z PLANE.                              DOC01780
C           ** DEFAULT = 0 **                                                 DOC01790
C       XXMAX,YYMAX,ZZMAX,XXMIN,YYMIN,ZZMIN LOCATE CUTTING PLANES             DOC01800
C           PARALLEL TO PRINCIPAL (X-Y,X-Z,Y-Z) PLANES                        DOC01810
C           TO LIMIT PLOT.                                                    DOC01820
C           ** DEFAULT XXMAX=YYMAX=ZZMAX=1.0E+20 **                           DOC01830
C           ** DEFAULT XXMIN=YYMIN=ZZMIN=-1.0E+20 **                          DOC01840
C       NDMAX = MAXIMUM GRID PT. TO BE INCLUDED IN PLOT.                      DOC01850
C           ** DEFAULT = 9999999999 **                                        DOC01860
C       NDMIN = MINIMUM GRID PT. TO BE INCLUDED IN PLOT.                      DOC01870
C           ** DEFAULT = 0 **                                                 DOC01880
C       NELMAX = MAXIMUM ELEMENT NUMBER TO BE INCLUDED IN PLOT.               DOC01890
C           ** DEFAULT = 9999999999 **                                        DOC01900
C       NELMIN = MINIMUM ELEMENT NUMBER TO BE INCLUDED IN PLOT.               DOC01910
C           ** DEFAULT = 0 **                                                 DOC01920
```

```
C        KODE SPECIFIES CONTROL OPTION AFTER PLOT IS COMPLETE.
C        KODE = 0, LAST PLOT, EXIT FROM PROGRAM.
C             = 1, READ ANOTHER NAMELIST PICT.
C             = 2, READ A NEW SET OF DISPLACEMENT DATA, INCLUDING A
C                  CASE IDENTIFICATION CARD IF PRESENT.
C             = 3, READ A COMPLETE NEW SET OF INPUT DATA,
C                  INCLUDING A TITLE CARD.
C             ** DEFAULT = 0 **


C    THE ABOVE COMPRISES A COMPLETE BASIC SET OF INPUT DATA IF
C    KODE = 0 IN $PICT.  FOR KODE = 1, 2, OR 3, ADDITIONAL SECTIONS OF
C    THE BASIC DECK MUST BE REPEATED.  THE DECK MUST END WITH
C    NAMELIST $PICT HAVING KODE = 0.
C
C
C
C**********************************************************************
C
C
C                   DESCRIPTION OF GRAPHICS SUBROUTINES
C
C
C
C          THE SUBROUTINES USED IN THE ACTUAL CREATION OF PLOTS BY
C          THE CALCOMP MODEL 765 CAN BE FOUND IN NPS TECHNICAL NOTE
C          NUMBER 0211-03, "PLOTTING PACKAGE FOR NPS IBM 360/367".
C
C
C
C*********************************************************************
      SUBROUTINE PSAP(ZZZ, NZ, DISPD, NON)
C
C *** THIS IS THE MAIN SUBROUTINE WHICH CALLS OTHER SUBROUTINES
C
      INTEGER NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT
      COMMON/CDATA/NTIME,NTLC
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KFORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      COMMON/LIMITS/ XXMAX,YYMAX,ZZMAX,XXMIN,YYMIN,ZZMIN,NDMAX,NDMIN,
     1NELMAX,NELMIN
      COMMON/CORGN/ XOABS,YOABS,XPMAX,XSPACE,PSIZE
      COMMON/GLOOP/ ILOOP
      COMMON/ABLK/ A(3,3)
      COMMON/SAVEV/ DMAGS,IDMAG
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
```

```
      CCMMON/VALUES/ NVALUS
      COMMON/CASEID/ IDCASE
      DIMENSION ZZZ(NZ),DISPD(5,3,NON)
      CIMENSION DSAV(3)
      REAL*8 ABCD(10)
      NAMELIST/PICT/ KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,
     1PLCTSZ,XORGN,YORGN,PSCALE,NOTAT,KDISP,IDMAG,DMAGS,KODE,
     2KSYMXY,KSYMXZ,KSYMYZ,XXMAX,YYMAX,ZZMAX,XXMIN,
     3YYMIN,ZZMIN,NDMAX,NDMIN,NELMAX,NELMIN,XLHT
C
C *** TC ZERO NODE AND ELEMENT SUMMATICN COUNTERS
C
      ILCOP = 0
      NNODE = 0
      XCABS = 0.0
      YCABS = 0.0
      XFMAX = 0.0
  500 CCNTINUE
      REWIND 10
      XSTRT = 0.0
      YSTRT = 0.0
      WRITE(6,8)
    8 FCRMAT(1H1)
C
C *** TO READ TITLE CARD FOR RUN
C
      READ(5,10,END=999) ABCD
   10 FCRMAT(10A8)
      WRITE(6,11) ABCD
   11 FCRMAT(///,20X,10A8,///)
      CALL INITAL
      HEIGHT = 0.15
      XSTRT = XSTRT+2.0*HEIGHT
      YSTRT = 1.0
      CALL NOTATE(XSTRT,YSTRT,HEIGHT,ABCD, 0.0,80)
      CALL CALPLT(-.5,0.0,-3)
C
C *** TO SET POINTERS FOR BLANK COMMON STORAGE ZZZ
C *** (WITH INTEGER NAMES OF ARRAYS USED IN CALLED SUBROUTINES)
C
      NUMPT = 1
      XPT = NUMPT+NNDEST
      YPT = XPT+NNDEST
      ZPT = YPT+NNDEST
      UPT = ZPT+NNDEST
      IF(NUDISP.EQ.0) VPT = UPT+1
      IF(NUDISP.NE.0) VPT = UPT+MNDEST
      IF(NVDISP.EQ.0) WPT = VPT+1
```

```
      IF(NVDISP.NE.0) WPT = VPT+NNDEST
      IF(NWDISP.EQ.0) NEND = WPT+1-1
      IF(NWDISP.NE.0) NEND = WPT+NNDEST-1
      WRITE(6,15) NEND
   15 FORMAT(///,20X,'BLANK COMMON STORAGE ZZZ REQUIRES AT LEAST ',I6,
     1' LOCATIONS FOR THIS CASE'///)
      IF(KGEOM.EQ.1) CALL GEOM1
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      IF(KGEOM.EQ.2) CALL GEOM2
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      IF(KGEOM.EQ.9) CALL GEOM9
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      CALL PNTOUT(1,
     1ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
  600 CONTINUE
      IF(IDCASE.EQ.0) GO TO 650
      READ(5,10) ABCD
      WRITE(6,11) ABCD
  650 CONTINUE
      CALL ZEROD
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      IF(KDATA.EQ.1) CALL DATA1
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      IF(KDATA.EQ.5) CALL DATA5
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      IF(KDATA.EQ.9) CALL DATA9
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT),
     2DISPD,NON)
      CALL PNTOUT(2,
     1ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
  700 CONTINUE
      IF(KPLOT.EQ.4.AND.ILOOP.NE.0) GO TO 6000
      WRITE (6,1000)
 1000 FORMAT(///)
      READ(5,PICT)
      WRITE(6,PICT)
 6000 CONTINUE
      CALL DSCALE
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      CALL BOUND
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      CALL ROTAT
      CALL PLOTX
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      IF(NOTAT.EQ.1) CALL NDLET
     1(ZZZ(NUMPT),ZZZ(XPT),ZZZ(YPT),ZZZ(ZPT),ZZZ(UPT),ZZZ(VPT),ZZZ(WPT))
      IF(KPLOT.EQ.4) CALL CALPLT( 0.0,0.0,-3)
      IF(KPLOT.EQ.4) CALL CCRT2
```

```
      ILOCP = ILOOP+1
      IF(KODE.EQ.0) GO TO 800
      GO TO (700,600,500), KODE
  800 CONTINUE
  999 CALL PSTOP
      RETURN
      END
      SLBROUTINE PSTOP
C
C *** TC TERMINATE JOB.
C
      CCMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      CALL CALPLT(0.0,0.0,-3)
      CALL PLOTE
      STOP
      END
      SLBROUTINE INITAL
C
C *** TO SET LP VALUES FOR CONTROL PARAMETERS
C
      COMMON/CDATA/NTIME,NTLC
      CCMMCN/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      CCMMON/LIMITS/ XXMAX,YYMAX,ZZMAX,XXMIN,YYMIN,ZZMIN,NDMAX,NDMIN,
     1NELMAX,NELMIN
      CCMMON/CORGN/ XOABS,YOABS,XPMAX,XSPACE,PSIZE
      CCMMON/SAVEV/ DMAGS,IDMAG
      CCMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWCISP
      CCMMON/SEQNCE/ IRESEQ
      CCMMCN/VALUES/ NVALUS
      COMMON/CASEID/ IDCASE
      NAMELIST/OPTION/ NNDEST,NUDISP,NVDISP,NWDISP,
     1KGECM,KDATA,NVALUS,IRESEQ,KPLOT,XSPACE,PSIZE,IDCASE
C
C *** DESCRIPTION OF VALUES IN $OPTION GIVEN IN SUBROUTINE DOCMNT
C
C
C *** TO SET DEFAULT VALUES FOR $OPTION
C
      NNDEST = 200
      NUDISP = 1
      NVDISP = 1
      NWDISP = 1
      KGECM=9
      KDATA=9
```

```
      NTIME=0
      NVALUS = 0
      IRESEQ = 1
      KPLOT = 1
      XSPACE=5.0
      PSIZE=10.0
      IDCASE = 0
C
C *** TO SET DEFAULT VALUES FOR $PICT
C
      KHORZ = 1
      KVERT = 2
      PHI = 0.0
      THETA = 0.0
      PSI = 0.0
      NEWFR = 1
      ISCALE = 1
      PLOTSZ = 10.0
      XORGN = 0.0
      YORGN = 0.0
      PSCALE = 1.0
      NOTAT = 0
      XLHT = 0.15
      KDISP = 0
      IDMAG = 2
      DMAGS = 1.0
      KSYMXY = 0
      KSYMXZ = 0
      KSYMYZ = 0
      XXMAX = 1.0E20
      YYMAX = 1.0E20
      ZZMAX = 1.0E20
      XXMIN = -1.0E20
      YYMIN = -1.0E20
      ZZMIN = -1.0E20
      NDMAX = 9999999
      NDMIN = 0
      NELMAX = 9999999
      NELMIN = 0
      KODE = 0
      READ(5,OPTION,END=999)
      IF(KPLOT.LE.2) CALL CALCMP
      RETURN
  999 CALL PSTOP
      RETURN
      END
      SUBROUTINE BOUND(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C
```

```
C *** TO DETERMINE MAXIMUM DIMENSIONAL LIMITS OF BODY FOR USE          DOC04330
C     IN SCALING PLOTS                                                  DOC04340
C                                                                       DOC04350
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT, DOC04360
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,        DOC04370
     2PSCALE,KDISP,DMAG,KODE                                            DOC04380
      COMMON/LIMITS/ XXMAX,YYMAX,ZZMAX,XXMIN,YYMIN,ZZMIN,NDMAX,NDMIN,   DOC04390
     1NELMAX,NELMIN                                                     DOC04400
      COMMON/XYZLIM/ XYZMAX(3),XYZMIN(3)                                DOC04410
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP                   DOC04420
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)      DOC04430
      DIMENSION NODE(4)                                                 DOC04440
      DO 5 I=1,3                                                        DOC04450
      XYZMIN(I) = +1.0E20                                               DOC04460
      XYZMAX(I) = -1.0E20                                               DOC04470
    5 CONTINUE                                                          DOC04480
      REWIND 10                                                         DOC04490
  100 CONTINUE                                                          DOC04500
      READ(10,END=1000) NUMEL,NODE1,NODE2,NODE3,NODE4                   DOC04510
      IF(NUMEL.LT.NELMIN.OR.NUMEL.GT.NELMAX) GO TO 100                  DOC04520
      NODE(1) = NODE1                                                   DOC04530
      NODE(2) = NODE2                                                   DOC04540
      NODE(3) = NODE3                                                   DOC04550
      NODE(4) = NODE4                                                   DOC04560
      DO 10 I=1,4                                                       DOC04570
      ND = NODE(I)                                                      DOC04580
      IF(NODE(I).EQ.0) GO TO 15                                         DOC04590
      IF(NUMPT(ND).LT.NDMIN.OR.NUMPT(ND).GT.NDMAX) GO TO 100            DOC04600
   10 CONTINUE                                                          DOC04610
   15 CONTINUE                                                          DOC04620
      DO 20 I=1,4                                                       DOC04630
      IF(NODE(I).EQ.0) GO TO 25                                         DOC04640
      ND = NODE(I)                                                      DOC04650
      IF(XPT(ND).GT.XXMAX) GO TO 20                                     DOC04660
      IF(XPT(ND).LT.XXMIN) GO TO 20                                     DOC04670
      IF(YPT(ND).GT.YYMAX) GO TO 20                                     DOC04680
      IF(YPT(ND).LT.YYMIN) GO TO 20                                     DOC04690
      IF(ZPT(ND).GT.ZZMAX) GO TO 20                                     DOC04700
      IF(ZPT(ND).LT.ZZMIN) GO TO 20                                     DOC04710
      IF(XPT(ND).GT.XYZMAX(1)) XYZMAX(1) = XPT(ND)                      DOC04720
      IF(XPT(ND).LT.XYZMIN(1)) XYZMIN(1) = XPT(ND)                      DOC04730
      IF(YPT(ND).GT.XYZMAX(2)) XYZMAX(2) = YPT(ND)                      DOC04740
      IF(YPT(ND).LT.XYZMIN(2)) XYZMIN(2) = YPT(ND)                      DOC04750
      IF(ZPT(ND).GT.XYZMAX(3)) XYZMAX(3) = ZPT(ND)                      DOC04760
      IF(ZPT(ND).LT.XYZMIN(3)) XYZMIN(3) = ZPT(ND)                      DOC04770
   20 CONTINUE                                                          DOC04780
   25 CONTINUE                                                          DOC04790
      GO TO 100                                                         DOC04800
```

```
 1000 CONTINUE
      DO 300 I=1,3
      IF(I.EQ.1.AND.KSYMYZ.NE.1) GO TO 300
      IF(I.EQ.2.AND.KSYMXZ.NE.1) GO TO 300
      IF(I.EQ.3.AND.KSYMXY.NE.1) GO TO 300
      XYZBIG = ABS(XYZMAX(I))
      IF(ABS(XYZMIN(I)).GT.XYZBIG) XYZBIG = ABS(XYZMIN(I))
      XYZMAX(I) = XYZBIG
      XYZMIN(I) = -XYZBIG
  300 CONTINUE
      RETURN
      END
      SUBROUTINE ZEROD(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C
C *** INITIALIZES ALL DISPLACEMENTS TO ZERO.
C
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
      IF(NUDISP.EQ.0) GO TO 200
      DO 150 I=1,NUDISP
      UPT(I) = 0.0
  150 CONTINUE
  200 CONTINUE
      IF(NVDISP.EQ.0) GO TO 300
      DO 250 I=1,NVDISP
      VPT(I) = 0.0
  250 CONTINUE
  300 CONTINUE
      IF(NWDISP.EQ.0) GO TO 400
      DO 350 I=1,NWDISP
      WPT(I) = 0.0
  350 CONTINUE
  400 CONTINUE
      RETURN
      END
      SUBROUTINE PNTOUT(IOUT,NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C
C *** FOR PRINTED OUTPUT OF INFORMATION IN BLANK COMMON - ZZZ
C
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
      GO TO (1000,2000), IOUT
 1000 CONTINUE
C
C *** FOR OUTPUT OF GEOMETRY INFORMATION
C
      WRITE(6,16)
   16 FORMAT(///,5X,'GRID POINT INFORMATION',///)
```

```
      WRITE(6,17)
   17 FORMAT(5X,'RESEQUENCED',4X,'USER INPUT'/
     15X,'GRIC POINT',5X,'GRID POINT'/
     25X,'NUMBER',9X,'NUMBER',13X,'X',14X,'Y',14X,'Z'//)
      DO 30 I=1,NNODE
      WRITE(6,18) I,NUMPT(I),XPT(I),YPT(I),ZPT(I)
   18 FORMAT(2X,I10,5X,I10,3X,3E15.4)
   30 CCNTINUE
      WRITE(6,19)
   19 FCRMAT(///,5X,'ELEMENT INFORMATION - WITH RESEQUENCED GRID POINTS
     1'///)
      WRITE(6,21)
   21 FORMAT(5X,'RESEQUENCED',4X,'USER INPUT',19X,'GRID POINTS'/
     15X,'ELEMENT',8X,'ELEMENT'/
     25X,'NUMBER',9X,'NUMBER',13X,'1',9X,'2',9X,'3',9X,'4'//)
      REWIND 10
      I = 0
   35 CCNTINUE
      I = I+1
      READ(10,END=999) NUMEL,NODE1,NODE2,NODE3,NODE4
      WRITE(6,22) I,NUMEL,NODE1,NODE2,NODE3,NODE4
   22 FCRMAT(2X,I10,5X,I10,4X,4I10)
      GO TO 35
 2000 CONTINUE
C
C *** FCR OUTPUT OF DISPLACEMENT DATA
C
      WRITE(6,210)
  210 FCRMAT(///,5X,'DISPLACEMENTS TO BE PLOTTED',///)
      WRITE(6,17)
      CO 230 I=1,NNODE
      U = 0.0
      IF(NUDISP.NE.0) U = UPT(I)
      V = 0.0
      IF(NVDISP.NE.0) V = VPT(I)
      W = 0.0
      IF(NWDISP.NE.0) W = WPT(I)
      WRITE(6,18) I,NUMPT(I),U,V,W
  230 CONTINUE
  999 RETURN
      END
      SLBROUTINE PLOTX(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C
C *** FCR GENERATING PLOTS.
C
      CCMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KFORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
```

```
      COMMON/LIMITS/ XXMAX,YYMAX,ZZMAX,XXMIN,YYMIN,ZZMIN,NDMAX,NDMIN,
     1NELMAX,NELMIN
      COMMON/XYZLIM/ XYZMAX(3),XYZMIN(3)
      COMMON/CORGN/ XOABS,YOABS,XPMAX,XSPACE,PSIZE
      COMMON/GLOOP/ ILOOP
      COMMON/ABLK/ A(3,3)
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
      COMMON/PDELS/ DELX,DELY
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
      DIMENSION NODE(4),X(4),Y(4),Z(4),XDISP(4),YDISP(4),ZDISP(4),
     1XROT(4),YROT(4)
    1 FORMAT(8A10)
    2 FORMAT(1X,8A10)
C
C *** TO MAKE ALL GRID POINT NUMBERS NEGATIVE
C
      DO 50 I=1,NNODE
      NUMPT(I) = -NUMPT(I)
   50 CONTINUE
      PI = 3.1415926
      XMOVE = 0.0
      IF(NEWFR.EQ.1) XMOVE = XPMAX+XSPACE
      YMOVE = -YOABS
      CALL CALPLT(YMOVE,XMOVE,-3)
      XOABS = XOABS+XMOVE
      YOABS = YOABS+YMOVE
      GO TO (701,701,703,701), KPLOT
  701 CONTINUE
      GO TO 710
  703 CONTINUE
      IF(NEWFR.EQ.1) CALL NFRAME
  710 CONTINUE
      DELX = 0.0
      DELY = 0.0
      IF(ISCALE.EQ.1) CALL XYSCAL
      CALL CALPLT(XORGN,YORGN,-3)
      XOABS = XOABS+XORGN
      YOABS = YOABS+YORGN
      XSHIFT = 0.0
      YSHIFT = 0.0
      ZSHIFT = 0.0
      XFMAX = -1.0E20
C
C *** LOOPS TO ACCOUNT FOR SYMMETRY
C
      ZSIGN = +1.0
      DO 500 II=1,2
      IF(II.EQ.2.AND.KSYMXY.NE.1) GO TO 500
```

```
      IF(II.EQ.2.AND.KSYMXY.EQ.1) ZSIGN = -1.0                          DOC06250
      YSIGN = +1.0                                                      DOC06260
      CC 510 JJ=1,2                                                     DOC06270
      IF(JJ.EQ.2.AND.KSYMXZ.NE.1) GO TO 510                             DOC06280
      IF(JJ.EQ.2.AND.KSYMXZ.EQ.1) YSIGN = -1.0                          DOC06290
      XSIGN = +1.0                                                      DOC06300
      CO 520 KK=1,2                                                     DOC06310
      IF(KK.EQ.2.AND.KSYMYZ.NE.1) GO TO 520                             DOC06320
      IF(KK.EQ.2.AND.KSYMYZ.EQ.1) XSIGN = -1.0                          DOC06330
C                                                                       DOC06340
C *** TC DETERMINE PROJECTED COORDINATES OF ELEMENTS                    DOC06350
C                                                                       DOC06360
      REWIND 10                                                         DOC06370
  100 CONTINUE                                                          DOC06380
      READ(10,END=1000) NUMEL,NODE1,NODE2,NODE3,NCDE4                   DOC06390
      IF(NUMEL.LT.NELMIN.OR.NUMEL.GT.NELMAX) GO TO 100                  DOC06400
      NODE(1) = NODE1                                                   DOC06410
      NCDE(2) = NODE2                                                   DOC06420
      NCDE(3) = NODE3                                                   DOC06430
      NCDE(4) = NODE4                                                   DOC06440
      DO 10 I=1,4                                                       DOC06450
      ND = NODE(I)                                                      DOC06460
      IF(NODE(I).EQ.0) GO TO 11                                         DOC06470
C                                                                       DOC06480
C *** TC MAKE GRID POINT NUMBERS CONNECTED BY ELEMENTS POSITIVE         DOC06490
      NUMPT(ND) = IABS(NUMPT(ND))                                       DOC06500
      IF(NUMPT(ND).LT.NDMIN.OR.NUMPT(ND).GT.NDMAX) GO TO 100            DOC06510
      NEND = I                                                          DOC06520
   10 CONTINUE                                                          DOC06530
   11 CONTINUE                                                          DOC06540
      I = KHORZ                                                         DOC06550
      J = KVERT                                                         DOC06560
      DO 20 N=1,NEND                                                    DOC06570
      ND = NODE(N)                                                      DOC06580
      IF(XPT(ND).GT.XXMAX) GO TO 100                                    DOC06590
      IF(XPT(ND).LT.XXMIN) GO TO 100                                    DOC06600
      IF(YPT(ND).GT.YYMAX) GO TO 100                                    DOC06610
      IF(YPT(ND).LT.YYMIN) GO TO 100                                    DOC06620
      IF(ZPT(ND).GT.ZZMAX) GO TO 100                                    DOC06630
      IF(ZPT(ND).LT.ZZMIN) GO TO 100                                    DOC06640
      XDISP(N) = 0.0                                                    DOC06650
      YDISP(N) = 0.0                                                    DOC06660
      ZDISP(N) = 0.0                                                    DOC06670
      IF(KDISP.EQ.1.AND.NUDISP.NE.0) XDISP(N) = UPT(ND)                 DOC06680
      IF(KDISP.EQ.1.AND.NVDISP.NE.0) YDISP(N) = VPT(ND)                 DOC06690
      IF(KDISP.EQ.1.AND.NWDISP.NE.0) ZDISP(N) = WPT(ND)                 DOC06700
      X(N) = XSIGN*(XPT(ND)+XDISP(N)*DMAG+XSHIFT)/PSCALE                DOC06710
      Y(N) = YSIGN*(YPT(ND)+YDISP(N)*DMAG+YSHIFT)/PSCALE                DOC06720
```

```
      Z(N) = ZSIGN*(ZPT(ND)+ZDISP(N)*DMAG+ZSHIFT)/PSCALE                DOC06730
   20 CONTINUE                                                          DOC06740
      IF(KDISP.EQ.2) CALL XPLOD(NEND,X,Y,Z)                             DOC06750
      XCENT = 0.0                                                       DOC06760
      YCENT = 0.0                                                       DOC06770
      DC 25 N=1,NEND                                                    DOC06780
      XROT(N) = A(I,1)*X(N)+A(I,2)*Y(N)+A(I,3)*Z(N)                     DOC06790
      YROT(N) = A(J,1)*X(N)+A(J,2)*Y(N)+A(J,3)*Z(N)                     DOC06800
      XCENT = XCENT+XROT(N)                                             DOC06810
      YCENT = YCENT+YROT(N)                                             DOC06820
      XROT(N) = XROT(N)+DELX                                            DOC06830
      YROT(N) = YROT(N)+DELY                                            DOC06840
      IF(XROT(N).GT.XPMAX) XPMAX = XROT(N)                              DOC06850
   25 CONTINUE                                                          DOC06860
      XCENT = XCENT/FLOAT(NEND)-(6.0/7.0)*XLHT                          DOC06870
      YCENT = YCENT/FLOAT(NEND)-XLHT/2.0                                DOC06880
      XCENT = XCENT+DELX                                                DOC06890
      YCENT = YCENT+DELY                                                DOC06900
      AL = NUMEL                                                        DOC06910
      IF(NOTAT.EQ.2) CALL NUMBER(XCENT,YCENT,XLHT,AL,0.0,-1)            DOC06920
C                                                                       DOC06930
C *** TC PLOT ELEMENTS                                                  DOC06940
C                                                                       DOC06950
      CALL CALPLT(XROT(1),YROT(1),3)                                    DOC06960
      DO 30 N=2,NEND                                                    DOC06970
      CALL CALPLT(XROT(N),YROT(N),2)                                    DOC06980
   30 CONTINUE                                                          DOC06990
      CALL CALPLT(XROT(NEND),YROT(NEND),3)                              DOC07000
      IF(NEND.LE.2) GO TO 36                                            DOC07010
   35 CALL CALPLT(XROT(1),YROT(1),2)                                    DOC07020
      CALL CALPLT(XROT(1),YROT(1),3)                                    DOC07030
   36 CONTINUE                                                          DOC07040
      GO TO 100                                                         DOC07050
 1000 CONTINUE                                                          DOC07060
      IF(KDISP.NE.3) GO TO 650                                          DOC07070
  600 CONTINUE                                                          DOC07080
C                                                                       DOC07090
C *** TC PLOT VECTORS AT GRID POINTS                                    DOC07100
C                                                                       DOC07110
      DC 601 ND=1,NNODE                                                 DOC07120
      IF(NUMPT(ND).LE.0) GO TO 601                                      DOC07130
      IF(NUMPT(ND).LT.NDMIN.OR.NUMPT(ND).GT.NDMAX) GO TO 601            DOC07140
      IF(XPT(ND).GT.XYZMAX(1)) GO TO 601                                DOC07150
      IF(XPT(ND).LT.XYZMIN(1)) GO TO 601                                DOC07160
      IF(YPT(ND).GT.XYZMAX(2)) GO TO 601                                DOC07170
      IF(YPT(ND).LT.XYZMIN(2)) GO TO 601                                DOC07180
      IF(ZPT(ND).GT.XYZMAX(3)) GO TO 601                                DOC07190
      IF(ZPT(ND).LT.XYZMIN(3)) GO TO 601                                DOC07200
```

```
      X(1) = XSIGN*(XPT(ND)+XSHIFT)/PSCALE
      Y(1) = YSIGN*(YPT(ND)+YSHIFT)/PSCALE
      Z(1) = ZSIGN*(ZPT(ND)+ZSHIFT)/PSCALE
      XDISP(1) = 0.0
      YDISP(1) = 0.0
      ZDISP(1) = 0.0
      IF(NUDISP.NE.0) XDISP(1) = UPT(ND)
      IF(NVDISP.NE.0) YDISP(1) = VPT(ND)
      IF(NWDISP.NE.0) ZDISP(1) = WPT(ND)
      X(2) = XSIGN*(XPT(ND)+XDISP(1)*DMAG+XSHIFT)/PSCALE
      Y(2) = YSIGN*(YPT(ND)+YDISP(1)*DMAG+XSHIFT)/PSCALE
      Z(2) = ZSIGN*(ZPT(ND)+ZDISP(1)*DMAG+ZSHIFT)/PSCALE
      I = KHORZ
      J = KVERT
      DO 605 N=1,2
      XROT(N) = A(I,1)*X(N)+A(I,2)*Y(N)+A(I,3)*Z(N)
      YROT(N) = A(J,1)*X(N)+A(J,2)*Y(N)+A(J,3)*Z(N)
      XROT(N) = XROT(N)+DELX
      YROT(N) = YROT(N)+DELY
  605 CONTINUE
      XARW = 0.06
      YARW = XARW/3.0
      CALL GARROW(XROT(1),YROT(1),XROT(2),YROT(2),1,XARW,YARW)
  601 CONTINUE
  650 CONTINUE
  520 CONTINUE
  510 CONTINUE
  500 CONTINUE
      RETURN
      END
      SUBROUTINE DSCALE(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C
C *** THIS SUBROUTINE DETERMINES THE SCALE FACTOR FOR DISPLACEMENTS
C
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      COMMON/SAVEV/ DMAGS,IDMAG
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
      IF(KDISP.EQ.0.OR.KDISP.EQ.2) GO TO 10
      GO TO (10,20), IDMAG
   10 CONTINUE
      DMAG = DMAGS
      GO TO 30
   20 CONTINUE
      DMAX = 0.0
      DO 100 I=1,NNODE
```

```
      IF(NUDISP.EQ.0) GO TO 500
      IF(ABS(UPT(I)).GT.DMAX) DMAX = ABS(UPT(I))
  500 CONTINUE
      IF(NVDISP.EQ.0) GO TO 501
      IF(ABS(VPT(I)).GT.DMAX) DMAX = ABS(VPT(I))
  501 CONTINUE
      IF(NWDISP.EQ.0) GO TO 502
      IF(ABS(WPT(I)).GT.DMAX) DMAX = ABS(WPT(I))
  502 CONTINUE
  100 CONTINUE
      DMAG = DMAGS/DMAX
   30 CONTINUE
      RETURN
      END
      SUBROUTINE ROTAT
C
C *** SETS UP COEFFICIENTS OF ROTATION MATRIX
C
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      COMMON/ABLK/ A(3,3)
      PI = 3.1415926536
      SINPHI = SIN(PHI*PI/180.0)
      COSPHI = COS(PHI*PI/180.0)
      SINTHE = SIN(THETA*PI/180.0)
      COSTHE = COS(THETA*PI/180.0)
      SINPSI = SIN(PSI*PI/180.0)
      COSPSI = COS(PSI*PI/180.0)
      A(1,1) = COSTHE*COSPSI
      A(1,2) = COSPSI*SINTHE*SINPHI-SINPSI*COSPHI
      A(1,3) = SINTHE*COSPHI*COSPSI+SINPHI*SINPSI
      A(2,1) = SINPSI*COSTHE
      A(2,2) = SINTHE*SINPHI*SINPSI+COSPHI*COSPSI
      A(2,3) = SINTHE*COSPHI*SINPSI-SINPHI*COSPSI
      A(3,1) = -SINTHE
      A(3,2) = COSTHE*SINPHI
      A(3,3) = COSTHE*COSPHI
      RETURN
      END
      SUBROUTINE XYSCAL
C
C *** TO DETERMINE SCALE FACTOR FOR MODEL GEOMETRY.
C
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      COMMON/XYZLIM/ XYZMAX(3),XYZMIN(3)
```

```
      COMMON/CORGN/ XOABS,YOABS,XPMAX,XSPACE,PSIZE
      COMMON/ABLK/ A(3,3)
      COMMON/PDELS/ DELX,DELY
      I = KHORZ
      J = KVERT
      DMAX = 0.0
      DO 5 N=1,3
      VDUM = ABS(XYZMAX(N)-XYZMIN(N))
      IF(VDUM.GT.DMAX) DMAX = VDUM
    5 CONTINUE
      PSCALE = DMAX/PLOTSZ
      DO 10 L=1,2
      DO 10 M=1,2
      DO 10 N=1,2
      X = XYZMIN(1)
      IF(L.EQ.2) X = XYZMAX(1)
      Y = XYZMIN(2)
      IF(M.EQ.2) Y = XYZMAX(2)
      Z = XYZMIN(3)
      IF(N.EQ.2) Z = XYZMAX(3)
      XROT = A(I,1)*X+A(I,2)*Y+A(I,3)*Z
      YROT = A(J,1)*X+A(J,2)*Y+A(J,3)*Z
      IF(L*M*N.NE.1) GO TO 30
   20 CONTINUE
      XRMIN = XROT
      XRMAX = XROT
      YRMIN = YROT
      YRMAX = YROT
   30 CONTINUE
      IF(XROT.GT.XRMAX) XRMAX = XROT
      IF(XROT.LT.XRMIN) XRMIN = XROT
      IF(YROT.GT.YRMAX) YRMAX = YROT
      IF(YROT.LT.YRMIN) YRMIN = YROT
   10 CONTINUE
      YR = ABS(YRMAX-YRMIN)
      IF(YR/PSCALE.GT.PSIZE) PSCALE = YR/PSIZE
      XRMAX = XRMAX/PSCALE
      YRMAX = YRMAX/PSCALE
      XRMIN = XRMIN/PSCALE
      YRMIN = YRMIN/PSCALE
      DELX = -XRMIN
      DELY = -YRMIN
      XORGN = 0.0
      YORGN = 0.0
      RETURN
      END
      SUBROUTINE XPLOD(NEND,X,Y,Z)
C
```

```
C *** FCR GENERATING EXPLODED PLOTS.
C
      CCMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      DIMENSION X(4),Y(4),Z(4)
C
C *** TO CALCULATE THE INCENTER OF TRIANGLES
C
      IF(NEND.NE.3) GO TO 20
   10 CONTINUE
      A = SQRT((X(2)-X(3))**2+(Y(2)-Y(3))**2+(Z(2)-Z(3))**2)
      B = SQRT((X(1)-X(3))**2+(Y(1)-Y(3))**2+(Z(1)-Z(3))**2)
      C = SQRT((X(1)-X(2))**2+(Y(1)-Y(2))**2+(Z(1)-Z(2))**2)
      AC1 = A/(A+B+C)
      AC2 = B/(A+B+C)
      AC3 = C/(A+B+C)
      XOC = AC1*X(1)+AC2*X(2)+AC3*X(3)
      YOC = AC1*Y(1)+AC2*Y(2)+AC3*Y(3)
      ZOC = AC1*Z(1)+AC2*Z(2)+AC3*Z(3)
      GC TO 190
   20 CCNTINUE
C
C *** TO CALCULATE THE CENTROID OF RODS,BARS,AND CUADS
C
      XCC = 0.0
      YOC = 0.0
      ZCC = 0.0
      DO 100 I=1,NEND
      XOC = XCC+X(I)
      YOC = YOC+Y(I)
      ZCC = ZOC+Z(I)
  100 CCNTINUE
      XOC = XOC/FLOAT(NEND)
      YOC = YOC/FLOAT(NEND)
      ZOC = ZOC/FLOAT(NEND)
  190 CONTINUE
C
C *** TO REDUCE THE SIZE OF THE ELEMENT
C
      DO 200 I=1,NEND
      X(I) = X(I)*DMAG
      Y(I) = Y(I)*DMAG
      Z(I) = Z(I)*DMAG
  200 CCNTINUE
C
C *** TO CALCULATE THE CENTROID OF THE REDUCED ELEMENT
C
```

```
      XRC = XCC*DMAG
      YRC = YOC*DMAG
      ZRC = ZOC*DMAG
C
C *** SHIFT CORNERS OF ORIGINAL AND RECUCED TO MAKE CENTROIDS MATCH
C
      DO 400 I=1,NEND
      X(I) = X(I)+(XOC-XRC)
      Y(I) = Y(I)+(YOC-YRC)
      Z(I) = Z(I)+(ZOC-ZRC)
  400 CONTINUE
      RETURN
      END
      SUBROUTINE GARROW(X1,Y1,X2,Y2,NC,XHEAD,YHEAD)
C
C *** TC DRAW ARROWS FROM X1,Y1 TO X2,Y2.
C
      DEN = SQRT((X2-X1)**2+(Y2-Y1)**2)
      IF(DEN.EQ.0.0) GO TO 5000
      C = (X1-X2)/DEN
      S = (Y1-Y2)/DEN
      CALL CALPLT(X1,Y1,3)
      CALL CALPLT(X2,Y2,2)
      IF(NC.LT.1) GO TO 1000
      XA = X2+(C*XHEAD-S*YHEAD)
      YA = Y2+(S*XHEAD+C*YHEAD)
      CALL CALPLT(XA,YA,2)
      IF(NC.LT.2) GO TO 1000
      XB = X2+(C*XHEAD-S*(-YHEAD))
      YB = Y2+(S*XHEAD+C*(-YHEAD))
      CALL CALPLT(XB,YB,2)
      IF(NC.LT.3) GO TO 1000
      CALL CALPLT(X2,Y2,2)
      IF(NC.LT.4) GO TO 1000
      XC = X2+(-S*YHEAD)
      YC = Y2+(+C*YHEAD)
      CALL CALPLT(XC,YC,2)
      IF(NC.LT.5) GO TO 1000
      XD = X2+(-S*(-YHEAD))
      YD = Y2+(+C*(-YHEAD))
      CALL CALPLT(XD,YD,2)
 1000 CONTINUE
      CALL CALPLT(X2,Y2,3)
 5000 CONTINUE
      RETURN
      END
      SUBROUTINE NDLET(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C
```

```
C *** FOR ANNOTATING GRID POINT NUMBERS ON PLOTS.
C
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      COMMON/LIMITS/ XXMAX,YYMAX,ZZMAX,XXMIN,YYMIN,ZZMIN,NDMAX,NDMIN,
     1NELMAX,NELMIN
      COMMON/XYZLIM/ XYZMAX(3),XYZMIN(3)
      COMMON/ABLK/ A(3,3)
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
      COMMON/PDELS/ DELX,DELY
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
      II = KHORZ
      JJ = KVERT
      XSHIFT = 0.0
      YSHIFT = 0.0
      ZSHIFT = 0.0
      DO 500 I=1,NNODE
      IF(NUMPT(I).LE.0) GO TO 500
      IF(NUMPT(I).LT.NDMIN.OR.NUMPT(I).GT.NDMAX) GO TO 500
      IF(XPT(I).GT.XYZMAX(1)) GO TO 500
      IF(XPT(I).LT.XYZMIN(1)) GO TO 500
      IF(YPT(I).GT.XYZMAX(2)) GO TO 500
      IF(YPT(I).LT.XYZMIN(2)) GO TO 500
      IF(ZPT(I).GT.XYZMAX(3)) GO TO 500
      IF(ZPT(I).LT.XYZMIN(3)) GO TO 500
      X = (XPT(I)+XSHIFT)/PSCALE
      Y = (YPT(I)+YSHIFT)/PSCALE
      Z = (ZPT(I)+ZSHIFT)/PSCALE
      XROT = A(II,1)*X+A(II,2)*Y+A(II,3)*Z
      YROT = A(JJ,1)*X+A(JJ,2)*Y+A(JJ,3)*Z
      XL = XROT+XLHT/2.0
      YL = YROT+XLHT/2.0
      XL = XL+DELX
      YL = YL+DELY
      AL = NUMPT(I)
      CALL NUMBER(XL,YL,XLHT,AL,0.0,-1)
  500 CONTINUE
      RETURN
      END
      SUBROUTINE DATA1(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
      RETURN
      END
      SUBROUTINE DATA5(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
      RETURN
      END
      SUBROUTINE GEOM1(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
      RETURN
```

```
      END
      SUBROUTINE GEOM2(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
      RETURN
      END
      SUBROUTINE NFRAME
      RETURN
      END
      SUBROUTINE CCRT2
      RETURN
      END
C
C ***** ADAPT FOR NPS SYSTEM
C
      SUBROUTINE CALCMP
      COMMON/PLOTC/ IBUFF(1024)
      CALL PLOTS
      RETURN
      END
C
C **** ADAPT FOR NPS SYSTEM
C
      SUBROUTINE CALPLT(X,Y,IPEN)
      CALL PLOT(X,Y,IPEN)
      RETURN
      END
C
C**** ADAPT FOR NPS SYSTEM
C
      SUBROUTINE NOTATE(X,Y,HT,BCD,THETA,N)
      DIMENSION BCD(1)
      CALL SYMBOL(X,Y,HT,BCD,THETA,N)
      RETURN
      END
      SUBROUTINE GEOM9(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT)
C
C *** USER SUPPLIED GEOMETRY INPUT SUBROUTINE.
C
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT,
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,
     2PSCALE,KDISP,DMAG,KODE
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP
      COMMON/GCONT/NUMNP,NPAR(14),NELTYP,NUMEL
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)
C
C *** INSERT ROUTINE HERE
C
      READ(5,100) HED
  100 FORMAT(12A6)
```

```
C                                                                       DOC10570
C*********READ MASTER CONTROL DARD                                      DOC10580
C                                                                       DOC10590
      READ(5,200)NUMNP,NELTYP                                           DOC10600
  200 FORMAT(2I5)                                                       DOC10610
      NNODE=NUMNP                                                       DOC10620
C                                                                       DOC10630
C*********READ OR GENERATE NODAL POINT DATA                             DOC10640
C                                                                       DOC10650
      NOLD=0                                                            DOC10660
   10 READ(5,300) N,XPT(N),YPT(N),ZPT(N),KN                             DOC10670
      NUMPT(N)= N                                                       DOC10680
  300 FORMAT(I5,30X,3F10.0,I5)                                          DOC10690
      IF (NOLD.EQ.0) GO TO 50                                           DOC10700
C                                                                       DOC10710
C*********CHECK IF GENERATION IS REQUIRED                               DOC10720
C                                                                       DOC10730
      IF (KN.EQ.0) GO TO 50                                             DOC10740
      NUM=(N-NOLD)/KN                                                   DOC10750
      NUMN=NUM-1                                                        DOC10760
      IF(NUMN.LT.1) GO TO 50                                            DOC10770
      XNUM=NUM                                                          DOC10780
      DX=(XPT(N)-XPT(NOLD))/XNUM                                        DOC10790
      DY=(YPT(N)-YPT(NOLD))/XNUM                                        DOC10800
      DZ=(ZPT(N)-ZPT(NOLD))/XNUM                                        DOC10810
      K=NOLD                                                            DOC10820
      DO 30 J=1,NUMN                                                    DOC10830
      KK=K                                                              DOC10840
      K=K+KN                                                            DOC10850
      XPT(K)=XPT(KK)+DX                                                 DOC10860
      YPT(K)=YPT(KK)+DY                                                 DOC10870
      ZPT(K)=ZPT(KK)+DZ                                                 DOC10880
      NUMPT(K)=K                                                        DOC10890
   30 CONTINUE                                                          DOC10900
   50 NOLD=N                                                            DOC10910
      IF(N.NE.NUMNP) GO TO 10                                           DOC10920
      CONTINUE                                                          DOC10930
      NUMEL=0                                                           DOC10940
C********* READ ELEMENT CONTROL CARDS                                   DOC10950
      DO 900 M=1,NELTYP                                                 DOC10960
      READ (5,1001,END=999) NPAR                                        DOC10970
 1001 FORMAT(14I5)                                                      DOC10980
      MTYPE=NPAR(1)                                                     DOC10990
      CALL ELTYPE(MTYPE)                                                DOC11000
  900 CONTINUE                                                          DOC11010
      ENDFILE 10                                                        DOC11020
  999 RETURN                                                            DOC11030
      END                                                               DOC11040
```

```
      SUBROUTINE ELTYPE(MTYPE)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12),MTYPE
    1 CALL TRUSS
      GO TO 900
    2 CALL BEAM
      GO TO 900
    3 CALL PLANE
      GO TO 900
    4 CALL PLANE
      GO TO 900
    5 CALL ERROR
      GO TO 900
    6 CALL SHELL
      GO TO 900
    7 CALL ERROR
      GO TO 900
    8 CALL ERROR
      GO TO 900
    9 CALL ERROR
      GO TO 900
   10 CALL ERROR
      GO TO 900
   11 CALL ERROR
      GO TO 900
   12 CALL ERROR
      GO TO 900
  900 RETURN
      END
      SUBROUTINE ERROR(MTYPE)
C
C *****   THIS SUBROUTINE TERMINATES THE PROGRAM CUE  ERROR IN INPUT
C
      WRITE(6,100) MTYPE
  100 FORMAT ('0ELEMENT TYPE',I4,'CANNOT BE PLOTTED')
      CALL PSTOP
      RETURN
      END
      SUBROUTINE TRUSS
C *** THIS SUBROUTINE READS TRUSS(ELTYPE 1) CONNECTIVITY
      COMMON/GCONT/NUMNP,NPAR(14),NELTYP,NUMEL
      NUME=NPAR(2)
      NUMMAT=NPAR(3)
C ****  READ MATERIAL PROPERTY CARDS (DUMMY)
      DO 10 I=1,NUMMAT
      READ(5,1001) DUMMY
 1001 FORMAT(10A8)
   10 CONTINUE
C ***** READ ELEMENT LOAD MUL. (DUMMY1)
```

```
      DO 20 I=1,4
      READ(5,1001) DUMMY1
   20 CONTINUE
      IF(NPAR(14).EQ.0) NPAR(14) = 1
      N= NPAR(14)
C *** READ ELEMENT CONNECTION INFORMATION OR GENERATE
  100 READ(5,1004) M,II,JJ,MTYP,TEM,KK
 1004 FORMAT(4I5,F10.0,I5)
      IF(KK.EQ.0) KK=1
  120 IF (M.NE.N) GO TO 200
      I=II
      J=JJ
      KKK=KK
  200 CONTINUE
      NUMEL=NUMEL+1
      K=0
      L=0
      WRITE (10)      N,I,J,K,L
      IF(N.EQ.NUME) RETURN
      N=N+1
      I=I+KKK
      J=J+KKK
      IF(N.GT.M) GO TO 100
      GO TO 120
      END
      SUBROUTINE PLANE
C
C ****  THIS SUBROUTINE READS MEMBRANE CARDS
C
      DIMENSION  EMUL(4,5),IE(5),IX(4)
      COMMON/GCONT/NUMNP,NPAR(14),NELTYP,NUMEL
      NUME= NPAR(2)
      NUMMAT= NPAR(3)
C ****  READ MATERIAL PROPERTIES
      DO 60 M=1,NUMMAT
      READ(5,1010) MAT,NT
1010  FORMAT(2I5)
      IF(NT.EQ.0) NT=1
      NTC=2* NT
      DO 50 K=1,NTC
      READ(5,1005) DUMMY
1005  FORMAT (10A8)
   50 CONTINUE
   60 CONTINUE
C
C****  READ ELEMENT LOAD FACTORS
C
      READ(5,1002) ((EMUL(I,J),J=1,5),I=1,4)
```

```
1002  FORMAT(5F10.0)                                                   DOC12010
C                                                                      DOC12020
C****  READ ELEMENT PROPERTIES                                         DOC12030
C                                                                      DOC12040
      IF(NPAR(14).EQ.0) NPAR(14) = 1                                   DOC12050
      N=NPAR(14)-1                                                     DOC12060
  130 READ(5,1003) M,(IE(I),I=1,4),KG                                  DOC12070
1003  FORMAT(5I5,30X,I5)                                               DOC12080
      IF(KG.EQ.0) KG=1                                                 DOC12090
      IF( IE(3).EQ.IE(4) )  IE(4)=0                                    DOC12100
  140 N=N+1                                                            DOC12110
      IF(M.EQ.N) GO TO 145                                             DOC12120
      DO 142 I=1,4                                                     DOC12130
  142 IX(I)=IX(I)+KG                                                   DOC12140
      GO TO 150                                                        DOC12150
  145 DO 148 I=1,4                                                     DOC12160
  148 IX(I)=IE(I)                                                      DOC12170
  150 CONTINUE                                                         DOC12180
      I = IX(1)                                                        DOC12190
      J = IX(2)                                                        DOC12200
      K = IX(3)                                                        DOC12210
      L = IX(4)                                                        DOC12220
      NUMEL=NUMEL+1                                                    DOC12230
      WRITE (10) N,I,J,K,L                                             DOC12240
  310 IF(N.EQ.NUME) RETURN                                             DOC12250
      IF(N.EQ.M) GO TO 130                                             DOC12260
      GO TO 140                                                        DOC12270
      END                                                              DOC12280
      SUBROUTINE BEAM                                                  DOC12290
C                                                                      DOC12300
C ***  THIS SUBROUTINE  READS BEAM(ELTYP 2) CONNECTIVITY               DOC12310
C                                                                      DOC12320
      COMMON/GCONT/NUMNP,NPAR(14),NELTYP,NUMEL                         DOC12330
      NUME=NPAR(2)                                                     DOC12340
      NUMEPC=NPAR(3)                                                   DOC12350
      NUMFEF=NPAR(4) * 2                                               DOC12360
      NUMMAT=NPAR(5)                                                   DOC12370
C ***  READ MATERIAL PROPERTY CARDS (DUMMY)                            DOC12380
      DO 10  I=1,NUMMAT                                                DOC12390
      READ(5,1001) DUMMY                                               DOC12400
 1001 FORMAT(10A8)                                                     DOC12410
   10 CONTINUE                                                         DOC12420
C ***  READ ELEMENT PROPERTY CARDS (DUMMY1)                            DOC12430
      DO 20  J=1,NUMEPC                                                DOC12440
      READ (5,1001) DUMMY1                                             DOC12450
   20 CONTINUE                                                         DOC12460
C ***  READ ELEMENT LOAD MULTIPLIERS(DUMMY2)                           DOC12470
      DO 30 K=1,3                                                      DOC12480
```

```
      READ (5,1001) DUMMY2                                              DOC12490
   30 CONTINUE                                                          DOC12500
C ***  READ  FIXED-END FORCE CARDS(DUMMY3)                              DOC12510
      DC 40 L=1,NUMFEF                                                  DOC12520
      READ(5,1001) DUMMY3                                               DOC12530
   40 CONTINUE                                                          DOC12540
      IF(NPAR(14).EQ.0) NPAR(14) = 1                                    DOC12550
      N=NPAR(14)                                                        DOC12560
C ***  READ ELEMENT CONNECTION  INFO                                    DOC12570
  100 READ(5,1002) M,II,JJ ,KK                                          DOC12580
 1002 FORMAT(3I5,47X,I8)                                                DOC12590
      IF (KK.EQ.0) KK=1                                                 DOC12600
  120 IF (M.NE.N) GO TO 200                                             DOC12610
      I = II                                                            DOC12620
      J = JJ                                                            DOC12630
      KKK = KK                                                          DOC12640
  200 CONTINUE                                                          DOC12650
      NUMEL = NUMEL+1                                                   DOC12660
      K = 0                                                             DOC12670
      L = 0                                                             DOC12680
      WRITE (10) N ,I,J,K,L                                             DOC12690
      IF (N.EQ.NUME) RETURN                                             DOC12700
      N = N + 1                                                         DOC12710
      I = I + KKK                                                       DOC12720
      J = J + KKK                                                       DOC12730
      IF (N.GT.M) GO TO 100                                             DOC12740
      GO TO 120                                                         DOC12750
      END                                                               DOC12760
      SUBROUTINE SHELL                                                  DOC12770
C                                                                       DOC12780
C **** THIS SUBROUTINE READS ELTYPE 6 CARDS                             DOC12790
C                                                                       DOC12800
      DIMENSION IY(7),IX(4)                                             DOC12810
      COMMON/GCONT/NUMNP,NPAR(14),NELTYP,NUMEL                          DOC12820
      ISTOP=0                                                           DOC12830
      NUME = NPAR(2)                                                    DOC12840
      NUMMAT = NPAR(3)                                                  DOC12850
      NMAT= 2*NUMMAT                                                    DOC12860
C **** READ MATERIAL PORPERTIES (DUMMY)                                 DOC12870
      DO 10 N=1,NMAT                                                    DOC12880
      READ(5,1000) DUMMY                                                DOC12890
 1000 FORMAT(10A8)                                                      DOC12900
   10 CONTINUE                                                          DOC12910
C **** READ ELEMENT LOAD FACTORS (DUMMY1)                               DOC12920
      DO 20 K=1,5                                                       DOC12930
      READ(5,1000) DUMMY1                                               DOC12940
   20 CONTINUE                                                          DOC12950
      IF(NPAR(14).EQ.0) NPAR(14) = 1                                    DOC12960
```

```
      NN = NPAR(14)-1                                                   DOC12970
  100 READ(5,1001) MM,IY                                                DOC12980
 1001 FCRMAT(8I5)                                                       DOC12990
  110 NN = NN + 1                                                       DOC13000
      IF (MM -NN) 440,50,60                                             DOC13010
   50 DO 45  I=1,7                                                      DOC13020
   45 IX(I) =  IY(I)                                                    DOC13030
      INCL = IY(7)                                                      DOC13040
      IF (INCL.EQ.0)  INCL=1                                            DOC13050
      GO TO 70                                                          DOC13060
   60 DO 65  I=1,4                                                      DOC13070
   65 IX(I)=  IX(I) + INCL                                              DOC13080
   70 CONTINUE                                                          DOC13090
      I=IX(1)                                                           DOC13100
      J= IX(2)                                                          DOC13110
      K=IX(3)                                                           DOC13120
      L=IX(4)                                                           DOC13130
      NUMEL = NUMEL +1                                                  DOC13140
      WRITE(10) NN,I,J,K,L                                              DOC13150
      GO TO 500                                                         DOC13160
  440 WRITE (6,2005) MM                                                 DOC13170
 2005 FCRMAT(19HOCARD FOR ELEMT(,I5,14H) IS IN ERROR.,1X)               DOC13180
      ISTOP = 1                                                         DOC13190
  500 IF(NN.LT.MM) GO TO 110                                            DOC13200
      IF(NN.EQ.NUME) RETURN                                             DOC13210
      IF(ISTOP.EQ.1) STOP                                               DOC13220
      GO TO 100                                                         DOC13230
      END                                                               DOC13240
      SUBROUTINE DATA9(NUMPT,XPT,YPT,ZPT,UPT,VPT,WPT,DISPD,NON)         DOC13250
C                                                                       DOC13260
C *** USER SUPPLIED DISPLACEMENT INPUT SUBROUTINE.                      DOC13270
C                                                                       DOC13280
      COMMON/CDATA/NTIME,NTLC                                           DOC13290
      COMMON/CONTRL/ KGEOM,KDATA,KPLOT,KSYMXY,KSYMXZ,KSYMYZ,NOTAT,XLHT, DOC13300
     1KHORZ,KVERT,PHI,THETA,PSI,NEWFR,ISCALE,PLOTSZ,XORGN,YORGN,        DOC13310
     2PSCALE,KDISP,DMAG,KODE                                            DOC13320
      COMMON/KOUNT/ NNODE,NNDEST,NUDISP,NVDISP,NWDISP                   DOC13330
      DIMENSION NUMPT(1),XPT(1),YPT(1),ZPT(1),UPT(1),VPT(1),WPT(1)      DOC13340
      DIMENSION DISPD(5,3,NON)                                          DOC13350
C                                                                       DOC13360
      IF(NUDISP.EQ.0) GO TO 25
      IF(NTIME.NE.0) GO TO 100                                          DOC13370
      READ(5,1000) NTLC,SCALEF                                          DOC13380
 1000 FORMAT(I5,F10.0)                                                  DOC13390
      IF(SCALEF.EQ.0) SCALEF=1.0                                        DOC13400
   10 READ(5,2000) N,NLCAS,U,V,W                                        DOC13410
 2000 FCRMAT(2I4,3E12.5)                                                DOC13420
      DISPD(NLCAS,1,N) = U*SCALEF                                       DOC13430
```

```
      DISPD(NLCAS,2,N) = V*SCALEF                                       DOC13440
      DISPD(NLCAS,3,N) = W*SCALEF                                       DOC13450
      IF( (NLCAS.EQ.NTLC).AND.(N.EQ. 1    ) ) GO TO 100                 DOC13460
      GO TO 10                                                          DOC13470
  100 NTIME = NTIME + 1                                                 DOC13480
  200 DO 20 I=1,NNODE                                                   DOC13490
      UPT(I) = DISPD(NTIME,1,I)                                         DOC13500
      VPT(I) = DISPD(NTIME,2,I)                                         DOC13510
      WPT(I) = DISPD(NTIME,3,I)                                         DOC13520
   20 CONTINUE                                                          DOC13530
   25 CONTINUE
      RETURN                                                            DOC13540
      END                                                               DOC13550
```

LIST OF FIGURES

LIST OF REFERENCES

1. Earthquake Engineering Research Center Report No. 73-11, SAP IV A Structural Analysis Program for Static and Dynamic Response of Linear Systems, by K. Bathe, E. L. Wilson, and F. E. Peterson, June 1973.

2. National Aeronautics and Space Adiministration TN D-7797, Digital Computer Programs for Generating Oblique Orthographic Projection and Contour Plots, by Gary L. Giles, January 1975.

3. Naval Postgraduate School Technical Note No. 0211-03, Plotting Package for NPS IBM 360/67, by Patricia C. Johnson, January 1974.

4. Naval Postgraduate School Technical Note No. 0211-05, User Libraries and Source Code Editing Under OS, by Sharon D. Raney, January 1976.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93940	2
4.	Assoc Professor R. E. Ball, Code 67 Bp Department of Aeronautics Naval Postgraduate School Monterey, California 93940	4
5.	Professor Gilles Cantin, Code 59 Ci Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	2
6.	Professor D. A. Collins, Code 67 Co Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1

7. Mr. Richard Citerley 2
Anamet Laboratories
P.O. Box 831
San Carlos, California 94070

8. LT Dennis M. Losh, USN 1
2583 E. 38th ST.
Lorain, Ohio 44055

Thesis 166998
L825 Losh
c.1
The implementation of a finite element computer code and associated pre- and postprocessor into AE4101 and AE4102 (Flight Vehicle Structural Analysis I and II).